



What's New in Flash 8

October 16, 2005

Josh Cavalier

Lodestone Digital, LLC

www.lodestone.com

josh@lodestone.com

Unit 1: Overview of the Session

Unit Objectives:

After this unit, you will:

- Describe the course objectives
- Know what the prerequisites of the class are
- Know the content of each unit for the course
- Understand the course materials

Unit Topics:

- About the Course
- Course Format
- Course Objectives
- Course Prerequisites
- Course Outline

About the Course

What's New in Flash 8 provides Macromedia Flash 8 users with the hands-on instruction that will help them become competent with all the new expressiveness features in Flash 8.

By the end of the course, students should be able to use what they learned here to build rich user experiences that include new features like complex gradients, blends, filters, Flash Video, and mobile device emulation.

The methodology for the course includes the following:

- **Clearly stated learning objectives**
Each unit in this course starts with the objectives for that unit.
- Presentation of material to support the learning objectives
- The instructor uses slides and other media to get across concepts.
- **Model uses of new material**
The instructors for this course demonstrate the use of the product.
- **Feedback from students**
The instructors for this course test for understanding during and after each unit.
- **Guided practice**
Each unit in this course has a series of walkthroughs that you perform along with your instructor so that you get familiar with new skills.
- **Independent practice**
Each unit in this course concludes with a hands-on lab that lets you practice a set of new skills, on your own, in context.

Course Format

This course is divided into 8 units, most of which present new information and contain demonstrations, walkthroughs and a lab where you can practice your new skills.

You will experience the following learning techniques to better understand the course:

- Concepts introduce new information.
- Demonstrations illustrate new concepts.
- Walkthroughs guide you, with the instructor's assistance, through procedures.
- Labs let you practice new skills on your own.
- Summaries provide a brief synopsis of the unit's content.

Course Objectives

After completing the course, you will be able to use the following new features in Flash 8:

- Toolbar Settings
- The Object Drawing Model
- Text and Graphic Rendering Options
- Ease In/Ease Out Animation Options
- Flash Video Components for Flash 8
- Blends and Filters
- Mobile Device Settings

Course Prerequisites

To get the most from this class, you should already be familiar with:

- The Windows XP operating system
- Basic Flash functionality including the interface, animation and ActionScript

Course Outline

This one day class contains the following 8 Units:

- Unit 1: Overview of the Session.
- Unit 2: Setting up your Flash 8 Document
- Unit 3: Interface Improvements
- Unit 4: Drawing Improvements
- Unit 5: Expressiveness Features and Animation
- Unit 6: Text and Graphics Rendering
- Unit 7: Using Video in Flash 8
- Unit 8: Mobile Phone Workflow

Unit 2: Setting Up the Flash 8 Document

Unit Objectives:

After this unit, you will be able to:

- Set up your document for web and mobile output
- Modify your document properties
- Define a Site in Flash 8
- Upload a Site via Flash 8
- Save versions of your FLA file

Unit Topics:

- Document Setup
- Document Setup for Flash Lite / Mobile Devices
- Defining a Site
- Uploading a Site
- Version Control

Setting Up Your Flash Document

In this unit, you will set up a simple Flash document, and examine some of the new features of setting up a FLA file.

Creating a Simple Flash Document

To illustrate the basic steps of creating any Flash document, this section guides you through the process in a simple tutorial.

The first step is to create a new document in Flash and explore one of the new features – SWF MetaData.

Setting Up a Document with SWF MetaData

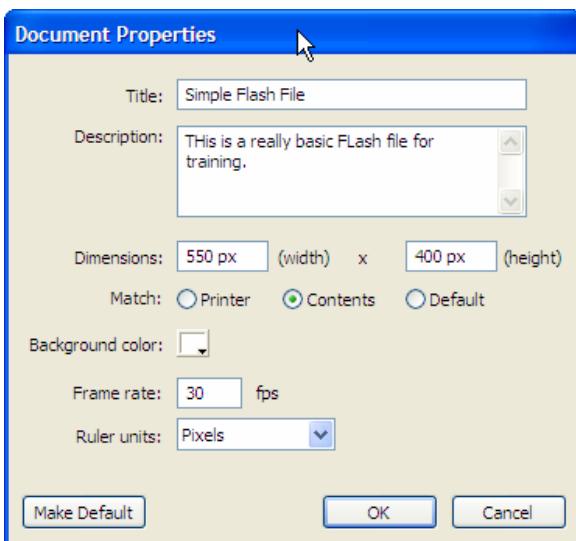
You can now add metadata to Flash files so that you can search more effectively for these files using Internet search engines, such as Google.com.

You can give your Flash content a search-engine friendly description and title. Search engines that look inside a SWF file can then better rank your Flash content.

To embed metadata within your SWF files, improving the ability of web-based search engines to return meaningful search results for Flash content, do the following:

- Enter a descriptive title in the Title text box. (Modify > Document...)
- Enter a description in the Description text box.

Descriptions can contain searchable keywords, author and copyright information, and short descriptions about the content and its purpose.



The search metadata is based on the RDF (Resource Description Framework) and XMP (Extensible Metadata Platform) specifications and is stored in Flash in a W3C-compliant format.

NOTE:

Flash lets you make the settings you specify in the Document Properties dialog box the default settings for any Flash document that you create. The exception to this is the Title and Description, which you need to specify for each Flash document that you create.

To set properties for a new or existing document in the Document Properties dialog box:

1. Select **File > New**.
2. With the document open, select **Modify > Document**.

The Document Properties dialog box appears.

3. To embed metadata within your SWF files, improving the ability of web-based search engines to return meaningful search results for Flash content, do the following:
 - Enter a descriptive title in the Title text box.
 - Enter a description in the Description text box.

Descriptions can contain searchable keywords, author and copyright information, and short descriptions about the content and its purpose.
4. The search metadata is based on the RDF (Resource Description Framework) and XMP (Extensible Metadata Platform) specifications and is stored in Flash in a W3C-compliant format.
5. For **Frame Rate**, enter 30 frames per second.
6. For **Dimensions**, specify the Stage size in pixels - enter values of 240 width and 270 height.
7. To set the background color of your document, click the triangle in the Background Color box and select a color from the palette. Pick any color you like!
8. To specify the unit of measure for rulers that you can display along the top and side of the application window, select an option from the pop-up menu in the upper right. We'll leave it pixels for this and future files.
9. To make the new settings the default properties for your new document only, click **OK**.
10. Save the file as **SimpleFlash.fla** in the Unit 2 folder.

Drawing a circle

After you've created your document, you are ready to add some artwork for the document.

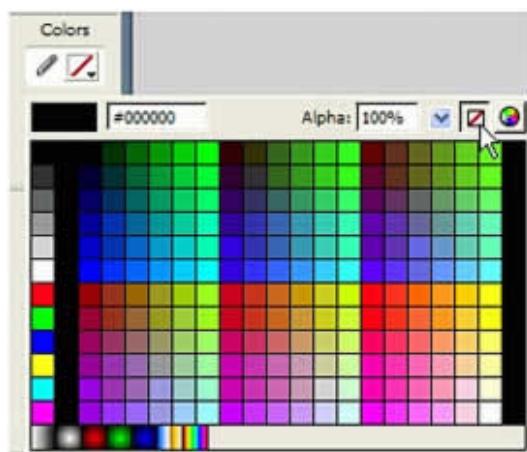
To draw a circle on the stage:

1. Select the Oval tool from the Tools panel.



The Oval tool in the Tools panel

2. Select the **No Color** option from the Stroke Color Picker.



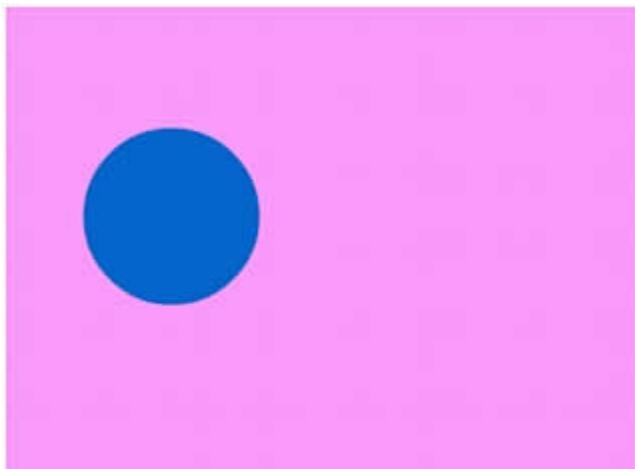
Selecting the No Color option in the Stroke Color Picker

3. Select a color of your choice from the Fill Color Picker.

Make sure the fill color contrasts well with the Stage color.

4. Draw a circle on the Stage by selecting the Oval tool and Shift-dragging on the Stage.

Holding the Shift key constrains the Oval tool to a circle.



The circle shape drawn on the Stage

Creating a symbol

You can turn your new artwork into a reusable asset by converting it to a Flash symbol. A symbol is a media asset that can be reused anywhere in your Flash document without the need to re-create it.

To create a symbol:

1. Click the Selection tool in the Tools panel.



The Tools panel with the Selection tool selected

2. Click the circle on the Stage to select it.

3. With the circle still selected, select **Modify > Convert to Symbol**.
4. In the Convert to Symbol dialog box, type **my_circle** into the Name text box.

The default behavior is now Movie Clip.

5. Click **OK**.

A square bounding box appears around the circle. You have now created a reusable asset, called a symbol, in your document.

6. The new symbol appears in the Library panel.

If the Library panel is not open, select **Window > Library**.

Animating the Circle

Now that you have some artwork in your document, you can make it more interesting by animating it to move across the Stage.

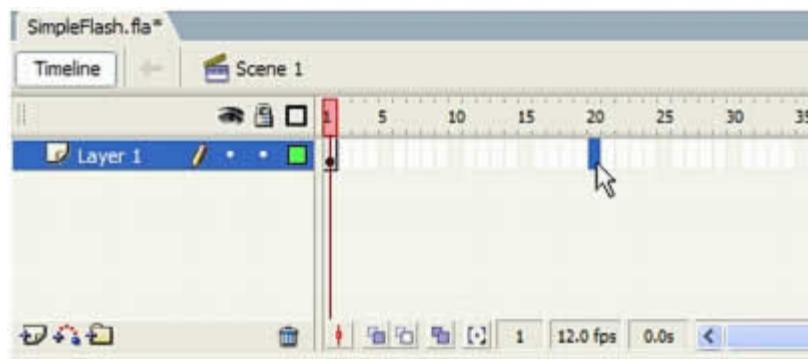
To create an animation with the circle:

1. Drag the circle to just left of the Stage area.



The circle shape moved to the left of the Stage area

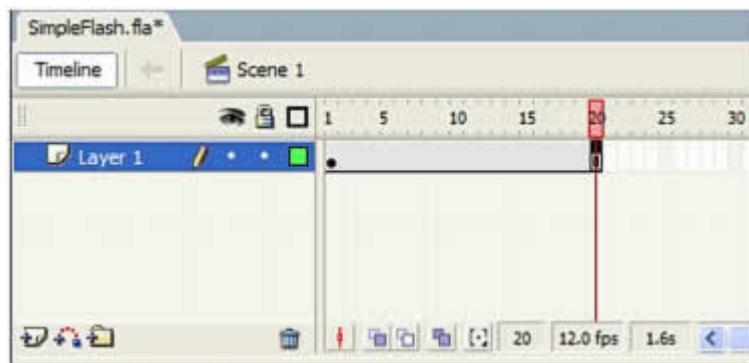
2. Click **Frame 20** of Layer 1 in the Timeline.



Selecting Frame 20 of Layer 1 in the Timeline

3. Select **Insert > Timeline > Frame**.

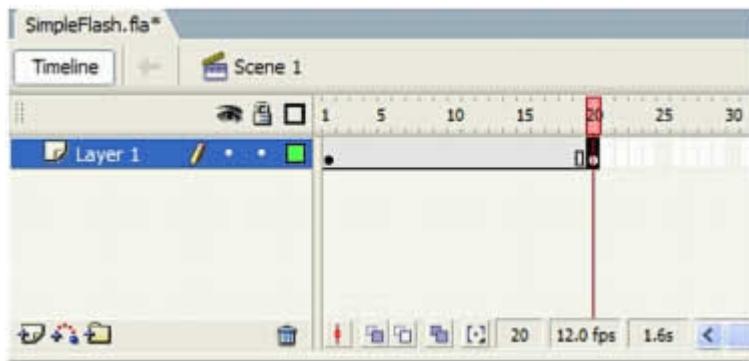
Flash adds frames to Frame 20, which remains selected.



Frames inserted in the Timeline

4. With Frame 20 still selected, select **Insert > Timeline > Keyframe**.

A keyframe is added in Frame 20. A keyframe is a frame where some property of an object is explicitly changed. In this new keyframe, you will change the circle's location.



Inserting a keyframe in Frame 20

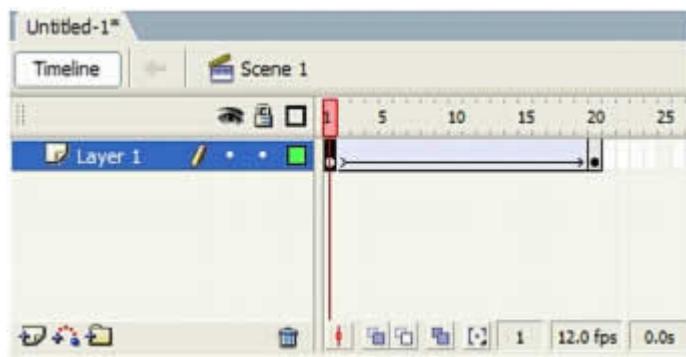
5. With Frame 20 still selected in the Timeline, drag the circle to just right of the Stage area.

6. Select Frame 1 of Layer 1 in the Timeline.
7. In the Property inspector (its default location is at the bottom of the Flash application window), select **Motion** from the Tween pop-up menu.



Selecting a motion tween in the Property inspector

An arrow appears in the Timeline in Layer 1 between Frame 1 and Frame 20.



The Timeline with an arrow indicating a motion tween

This step creates a *tweened animation* of the circle moving from its position in the first keyframe in Frame 1 to its new position in the second keyframe in Frame 20.

8. In the Timeline, drag the red playhead back and forth from frame 1 to frame 20 to preview the animation.
9. Select **File > Save**.
10. Select **Control > Test Movie** to test the FLA file.
11. Close the Test Movie window.

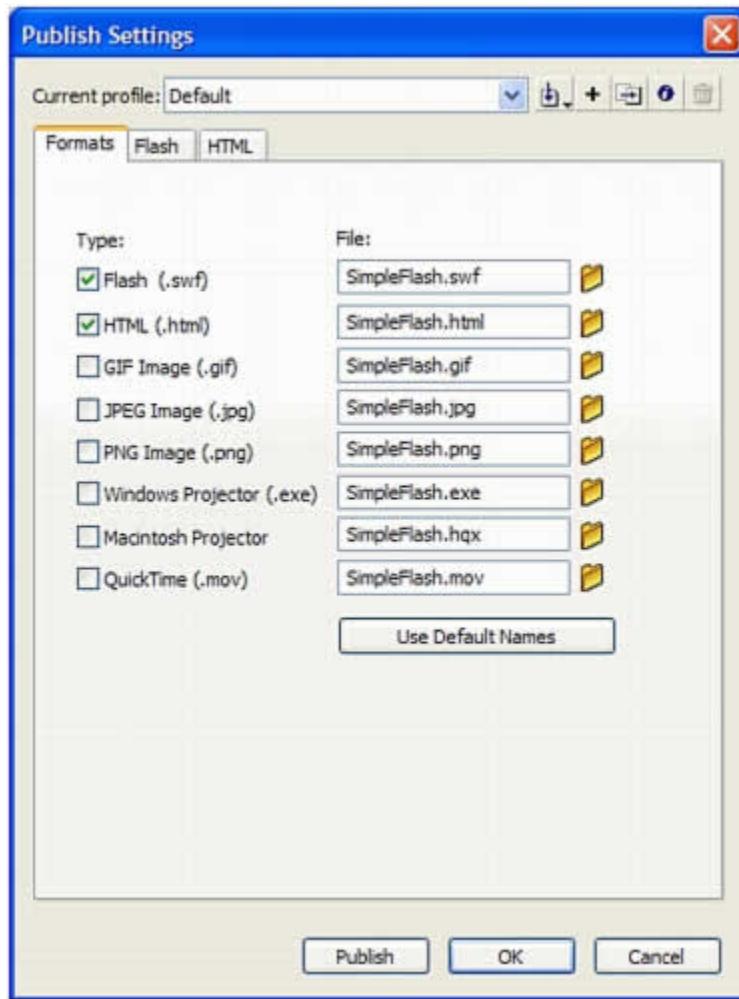
Publishing the File with HTML and Version Detection

When you finish your Flash document, you are ready to publish it so it can be viewed in a browser. When you publish a FLA file, Flash compresses it into the SWF file format. This is the format that you place in a web page. The Publish command can automatically generate an HTML file with the correct tags in it for you.

To publish the Flash file and view it in a browser:

1. Select **File > Publish Settings**.
2. In the Publish Settings dialog box, select the Formats tab and verify that only the Flash and HTML options are selected.

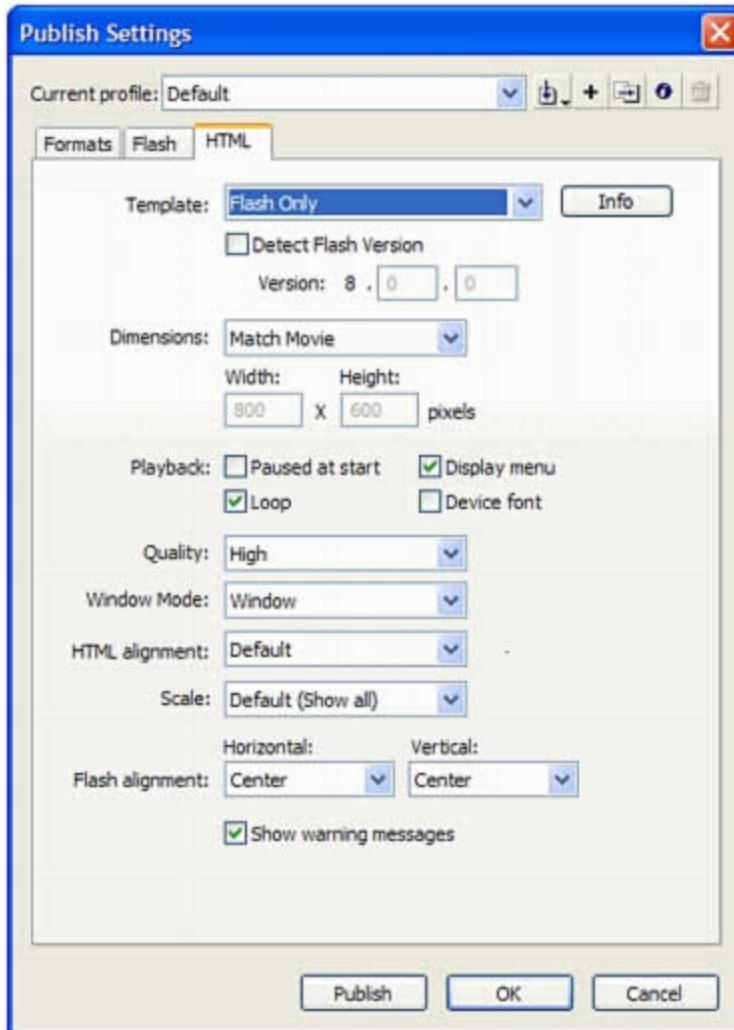
This action causes Flash to publish only the Flash SWF file and an HTML file. The HTML file is used to display the SWF file in a web browser.



The Flash and HTML options on the Formats tab

3. In the Publish Settings dialog box, select the HTML tab and verify that Flash Only is selected in the Template pop-up menu.

This template creates a simple HTML file that contains only your SWF file when displayed in a browser window.



Choosing Flash Only from the Template menu

4. Click **Detect Flash Version** – Keep the value at 8.0.0.
5. Click **OK**.
6. Select **File > Publish** and open your web browser.
7. Select **File > Open** in the web browser.
8. Navigate to the folder where you saved your FLA file in the Unit 2 folder.

The SimpleFlash.swf and SimpleFlash.html files are there. Flash creates these files when you click Publish.

9. Select the file named **SimpleFlash.html**.
10. Click **Open**. Your Flash document is displayed in the browser window. Congratulations! You have now completed your first Flash document.

Specifying Publish Settings with HTML and Version Detection

Let's go back and revisit our Detect Flash Version settings, and how Flash will publish your HTML template.

Playing Flash content in a web browser requires an HTML document that activates the SWF file and specifies browser settings. This document is generated automatically by the Publish command, from HTML parameters in a template document.

The template document can be any text file that contains the appropriate template variables-- including a plain HTML file, one that includes code for special interpreters such as ColdFusion or a template included with Flash.

You can customize a built-in template or manually enter HTML parameters for Flash using any HTML editor.

HTML parameters determine where the Flash content appears in the window, the background color, the size of the SWF file, and so on, and set attributes for the **object** and **embed** tags. You can change these and other settings in the HTML panel of the Publish Settings dialog box. Changing these settings overrides options you've set in the SWF file.

NOTE:

Flash Version Detection configures your document to detect the version of Flash Player that the user has and shows the user to alternate HTML code if the user does not have the targeted player.

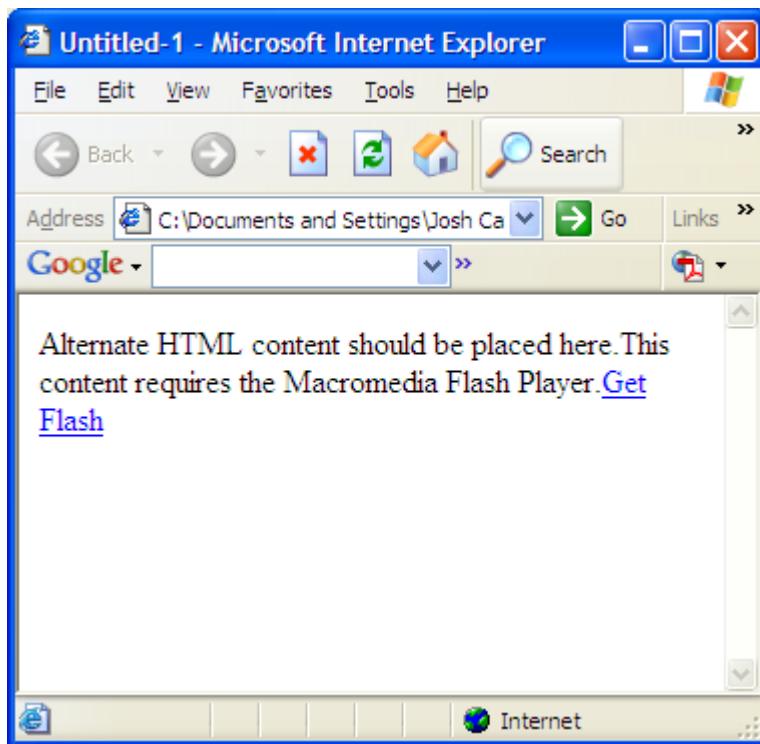
Currently, Flash uses JavaScript to detect the Flash player, and to display some simple HTML content. You will notice that the Content and Alternate file options have been eliminated from Flash 8.

For a more sophisticated experience you can download the Flash Detection Kit. This contains three ways of detecting the Flash player:

- ActionScript Detection – ActionScript solution
- Client-Side Detection – JavaScript/VB Script solution that comes with Flash 8
- Express Detection – Installs the player directly from the Macromedia website

You can find the kit at:

http://www.macromedia.com/software/flashplayer/download/detection_kit/



This is the window that appears if you use the JavaScript detection, and the user needs the Flash Player.

Detection Code

Let's take a close look at the source code that's used in our current document.

1. In the Browser select **View > Source** (IE) or **View > Page Source** (FireFox).
2. The alternate content that will be displayed is highlighted in the source code below. You can change this code to customize the user's experience.

```

<body bgcolor="#ffffff">
<!--url's used in the movie-->
<!--text used in the movie-->
<script language="JavaScript" type="text/javascript">
<!--
var hasRightVersion = DetectFlashVer(requiredMajorVersion,
requiredMinorVersion, requiredRevision);
if(hasRightVersion) { // if we've detected an acceptable version
  var oETags = '<object classid="clsid:D27CDB6E-AE6D-11cf-96B8-
444553540000" '
  + 'width="550" height="400"'
  +
'codebase="http://download.macromedia.com/pub/shockwave/cabs/flash/swflash.
cab">'
  + '<param name="movie" value="Untitled-2.swf" /><param name="quality"
value="high" /><param name="bgcolor" value="#ffffff" />'
  + '<embed src="Untitled-2.swf" quality="high" bgcolor="#ffffff" '
```

```

+ 'width="550" height="400" name="Untitled-2" align="middle"'
+ 'play="true"'
+ 'loop="false"'
+ 'quality="high"'
+ 'allowScriptAccess="sameDomain"'
+ 'type="application/x-shockwave-flash"'
+ 'pluginspage="http://www.macromedia.com/go/getflashplayer">'
+ '</embed>'
+ '</object>';
document.write(oeTags); // embed the flash movie
} else { // flash is too old or we can't detect the plugin
var alternateContent = 'Alternate HTML content should be placed here.'
+ 'This content requires the Macromedia Flash Player.'
+ '<a href="http://www.macromedia.com/go/getflash/">Get Flash</a>';
document.write(alternateContent); // insert non-flash content
}
// -->
</script>
<noscript>
// Provide alternate content for browsers that do not support
scripting
// or for those that have scripting disabled.
Alternate HTML content should be placed here. This content requires
the Macromedia Flash Player.
<a href="http://www.macromedia.com/go/getflash/">Get Flash</a>
</noscript>
</body>

```

3. Since the Flash 8 player has already been installed on your machine – you will not see the alternate content.
4. Close out of the browser and go back into Flash 8.

Document Setup for Mobile Devices

Flash Lite 1.0 and Flash Lite 1.1 are supported on a variety of mobile devices that are available in a number of different geographic regions and markets. Some of these devices are available globally, while others are available only in specific geographic regions or from specific mobile operators. Some devices come with Flash Lite pre-installed from the device manufacturer or mobile operator, while on others it can be installed after purchase.

NOTE:

For the most current list of devices that support Flash Lite, see the Supported Devices page located at www.macromedia.com/mobile/supported_devices/.

Globally available devices that support Flash Lite include the Symbian Series 60-based devices from Nokia, Sendo, and Siemens, and the Symbian UIQ-based devices from Sony-Ericsson. The stand-alone player installs as a "top-level" application that a user can start from the device's application menu (just like a text messaging application or a mobile web browser, for example).

Regionally available devices that support Flash Lite comprise a larger group of devices than are available globally. As of this writing, these devices are available primarily in Japan and come with Flash Lite pre-installed. On these devices, Flash Lite enables several different types of content, such as Flash screen savers or animated ring tones.

About Flash Lite Content Types

Before you start developing a Flash Lite application, you need to know the following:

- The device or devices on which the content will be running, or *target devices*. The Flash Lite player is installed on a variety of devices. For a full list of devices with Flash Lite installed, see the Supported Devices page on the Macromedia website at www.macromedia.com/mobile/supported_devices/.
- The Flash Lite content types supported by the target devices. Each Flash Lite installation supports one or more application modes, or *content types*. For example, some devices use Flash Lite to enable Flash-based screen savers or animated ring tones. Others use Flash Lite to render Flash content that is embedded in mobile web pages. Not all content types support all Flash Lite features.

Each Flash Lite content type, paired with a specific device, defines a specific set of Flash Lite features that are available to your application. For example, a Flash application that's running as a screen saver is not typically allowed to make network connections or download data.

The Flash Lite testing features in Macromedia Flash Professional 8 let you test against multiple devices and different Flash Lite content types. This lets you determine if your application uses features that aren't available for the type of content that you are developing.

Flash Lite authoring features in Flash Professional 8

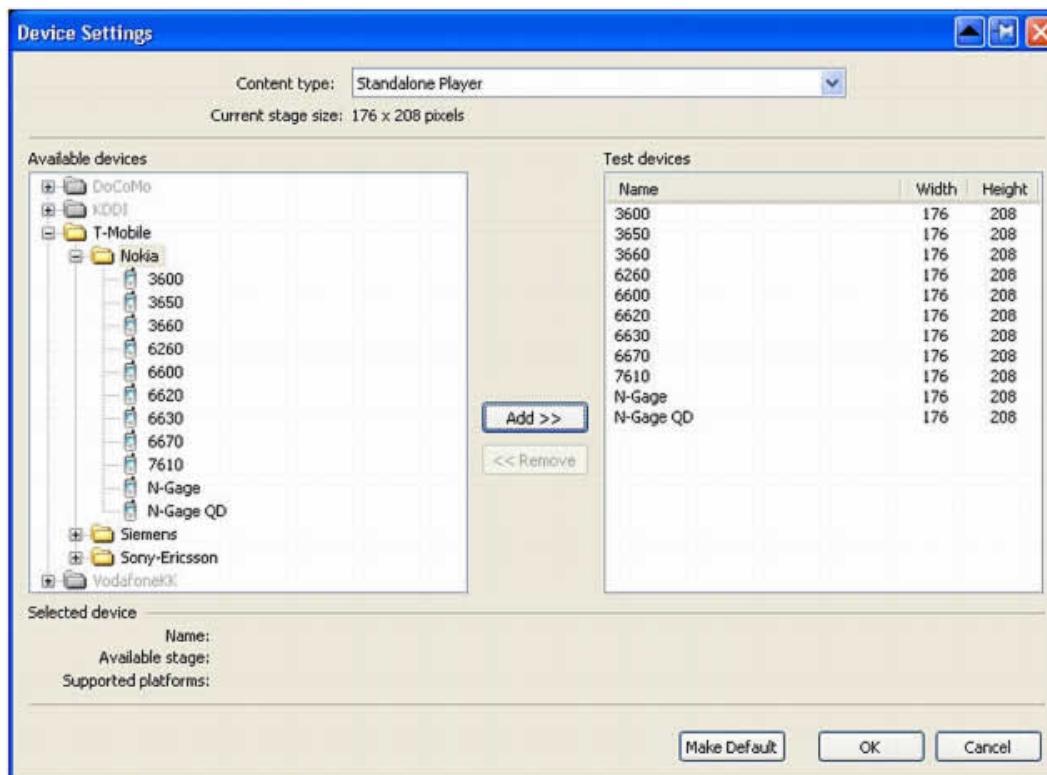
This section discusses the features in Flash Professional 8 designed specifically for Flash Lite developers. With the exception of the device templates feature, the following features are only available when your document's Version setting on the Flash tab of the Publish Setting dialog box is set to either Flash Lite 1.0 or Flash Lite 1.1.

The Flash Lite emulator and test window let you test your content as it will run and appear on an actual device. A settings pane in the Flash Lite test window lets you select a different test device, view information about your application, and set emulator debug output options.



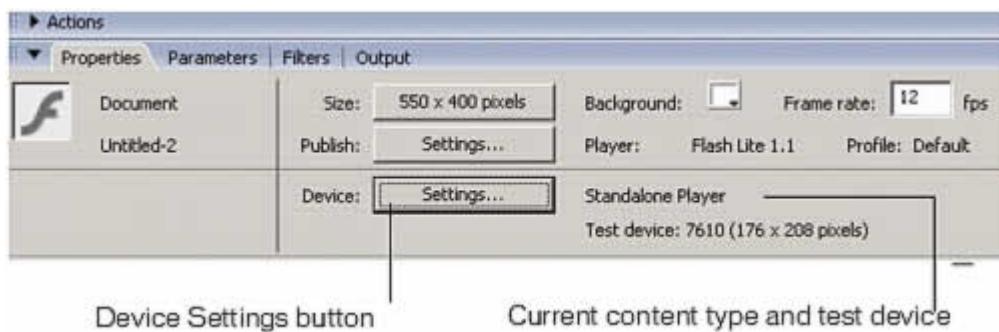
The Flash Lite emulator and test window.

The Device Settings dialog box lets you select the test devices and Flash Lite content type that you want to test against in the Flash Lite emulator. Different devices support different media types (for example, different types of device sound formats) as well as different Flash Lite content types, such as stand-alone player, screensaver, or browser. When you preview your application in the Flash Lite test window, the emulator mimics the features available to the selected test device running as the selected content type.



The Device Settings dialog box.

The **Property inspector** contains a section that provides information about your current device settings, as well as a button that lets you open the Device Settings dialog box. This button is active only when your document's Version setting on the Flash tab of the Publish Setting dialog box is set to Flash Lite 1.0 or Flash Lite 1.1.



Setting the Publishing Settings for Mobile Devices

1. Make sure you have the **SimpleFlash.fla** file still open.
2. In the Property inspector, click the **Device Settings** button. Because our current Publishing settings are set to Flash Player 8 you will see the following message:



3. Click the **Publishing Settings** button.
4. Select the **Flash** Tab.
5. Change your Flash player version to **Flash Lite 1.1**. Click OK. The Device setting dialog box appears.
6. Select **Browser** from the Content Type pop-up menu.
7. In the list of available devices, double-click the **NTT DoCoMo** folder to expand it, double-click the **Panasonic** folder.

8. With the **Panasonic** folder selected, click Add to add all the **Panasonic** devices to your list of test devices.
9. Test your application. **Control > Test Movie**. When you test your application in the emulator, you'll be able to test your application against any of your test devices for the selected content type.
10. Close the emulator window.

Saving and Version Control

When you save your FLA files, it is important to consider using a consistent naming scheme for your documents. This is particularly important if you save multiple versions of a single project.

Use intuitive, easy-to-read names for your files. Do not use spaces, capitalization, or special characters. Only use letters, numbers, dashes, and underscores. If you save multiple versions of the same file, devise a consistent numbering system such as site_menu01.swf, site_menu02.swf, and so on. Many designers and developers choose to use all lowercase characters in their naming schemes. Many Flash designers and developers adopt a naming system that uses a noun-verb or adjective-noun combination for naming files. Two examples of naming schemes are as follows: class_planning.swf and my_project.swf. Avoid cryptic file names.

It is good practice to save new versions of a FLA file when you build an extensive project. The following are different ways that you can save new versions of files:

Select File > Save As, and save a new version of your document.

Use version control software (such as SourceSafe, CVS, or Subversion) to control your Flash documents.

NOTE:

SourceSafe on Windows is the only supported version control software that integrates with the Project panel. You can use other version control software packages with FLA documents, but not necessarily in the Project panel.

Some problems might occur if you work with only one FLA file and do not save versions during the process of creating the file. It is possible that files might bloat in size because of the history that's saved in the FLA file or that a file might become corrupt (as with any software you use) while you are working on it. If any of these unfortunate events occur, you have other versions of your file to use if you save multiple versions throughout your development.

You might also encounter problems when you create an application. Perhaps you made a series of changes to the file and you do not want to use these changes. Or you might delete parts of the file that you want to use later in your development. If you save multiple versions while developing, you have an earlier version available if you need to revert.

There are several options that you can use to save a file: Save, Save As, and Save and Compact. When you save a file, Flash does not analyze all the data before creating an optimized version of the document. Instead, the changes you make to the document are appended to the end of the FLA file's data, which shortens the time it takes to save the document. When you select Save As and

type a new name for the file, Flash writes a new and optimized version of the file, which results in a smaller file size. When you select Save and Compact, Flash creates a new optimized file and deletes the original file.

Caution When Saving

When you select Save and Compact, you cannot undo any changes you made before you saved the file. If you select Save when working with a document, you can undo changes made prior to that save point. Because Save and Compact deletes the earlier version of the file and replaces it with the optimized version, you cannot undo earlier changes.

Remember to use **Save As** frequently and to type a new file name for your document after every milestone in your project if you are not using version control software to create backups of your FLA file. If you encounter major problems while working on the document, you have an earlier version to use instead of losing everything.

There are many software packages that let users use version control with their files, which enables teams to work efficiently and reduce errors (such as overwriting files or working on old versions of a document). Popular version control software programs include CVS, Subversion, and SourceSafe. As with other documents, you can use these programs to organize the Flash documents outside Flash.

Using Version Control with Projects

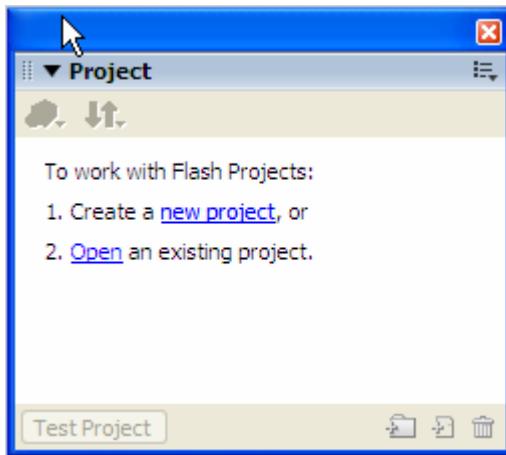
Version control in Flash Professional 8 lets you ensure that each author working in a project file is always using the latest version of a file, and that multiple authors do not overwrite each other's work.

To use version-control features, you must define a site for the project. You can specify a local, network, or FTP connection, or you can specify custom plug-ins for version control systems.

On Windows, you can use Flash projects with SourceSafe. You must have Microsoft Visual SourceSafe Client version 6 installed.

To define a site for version control:

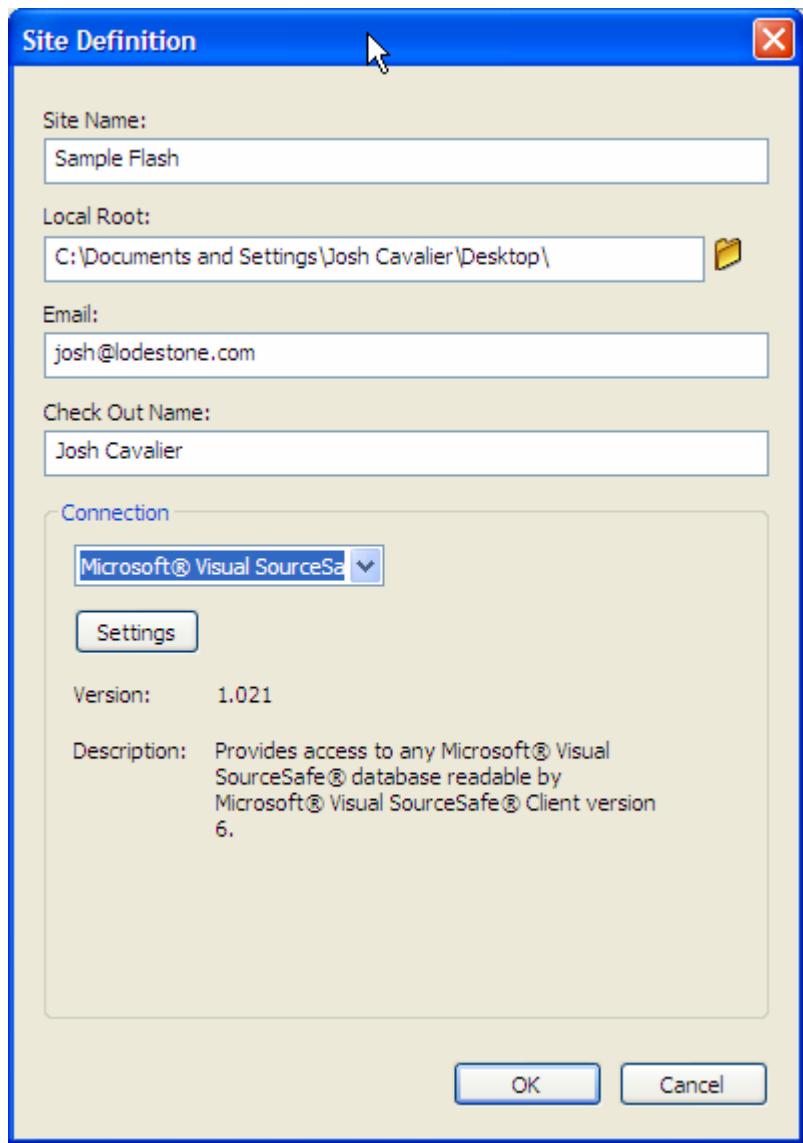
1. Create a new project (Window > Project) and save it as SimpleFlash.flp in the Unit 2 folder.
2. Add the SimpleFlash.fla, SimpleFlsh.swf, and SimpleFlash.html files from the Unit 2 folder.



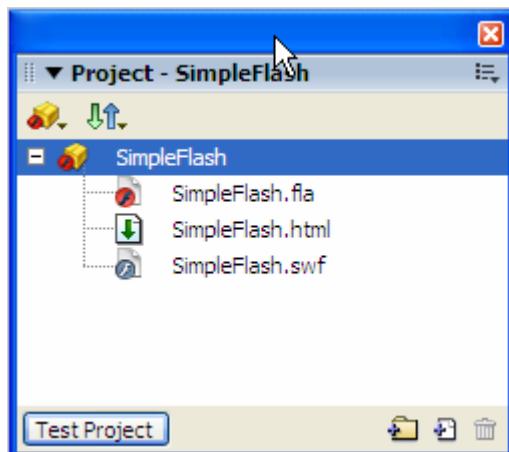
3. Select File > Edit Sites.
4. In the Edit Sites dialog box, click New.
5. In the Site Definition dialog box, enter the site name, the local root path, and the e-mail address and name of the user.
6. To specify a local, network, or FTP connection, select Local/Network or FTP from the Connection menu. Enter the location information for the Local/Network path or for the FTP connection and skip the next step.
7. To specify a Visual SourceSafe database, select SourceSafe Database from the Connection menu. Because we do not have VSS installed – just click OK.

NOTE:

SourceSafe database support is available for Windows only. You must have Microsoft Visual SourceSafe Client version 6 installed.



8. In the Flash Project panel (Window > Project), select Settings from the Project pop-up menu or context menu.



9. In the Project Settings dialog box, select the site definition from the Site menu in the Version Control section. Click OK.
10. In the Project pop-up menu, select Check In. Flash checks all files in the current project into the site.

LAB

- Create a new Flash movie
- Add a motion tween
- Set up the Publishing Setting for any Sony cell phone.

New Features Summary

- **SWF file metadata** You can now add metadata to Flash files so that you can search more effectively for these files using Internet search engines, such as Google.com.
- **Flash Player Detection.** Currently, Flash 8 uses JavaScript to detect the Flash player, and to display some simple HTML content. You will notice that the Content and Alternate file options have been eliminated from Flash 8.

For a more sophisticated experience you can download the Flash Detection Kit. This contains three ways of detecting the Flash player:

- ActionScript Detection – ActionScript solution
- Client-Side Detection – JavaScript/VB Script solution that comes with Flash 8
- Express Detection – Installs the player directly from the Macromedia website

You can find the kit at:

http://www.macromedia.com/software/flashplayer/download/detection_kit/

- **The Flash Lite emulator and test window** let you test your content as it will run and appear on an actual device. A settings pane in the Flash Lite test window lets you select a different test device, view information about your application, and set emulator debug output options.
- **The Property inspector** contains a section that provides information about your current device settings, as well as a button that lets you open the Device Settings dialog box. This button is active only when your document's Version setting on the Flash tab of the Publish Setting dialog box is set to Flash Lite 1.0 or Flash Lite 1.1.

Unit 3: Interface Improvements

Unit Objectives:

After this unit, you will be able to:

- Describe the new interface features in Flash 8
- Use the new Property Inspector
- Work with the Expanded Stage Area
- Customize the Flash Toolbar
- Use the new Preferences panel
- Use the new Library Panel and Tabbed Panels

Unit Topics:

- Working with the Property Inspector
- Viewing the Expanded Stage Area
- Customizing the Flash Toolbar
- Modifying the Property Inspector
- Changing Preferences
- Using the Flash 8 Library Panel
- Using Tabbed Panels

Working with the Property Inspector

The Property inspector simplifies document creation by making it easy to access the most commonly used attributes of the current selection, either on the Stage or in the Timeline. You can make changes to the object or document attributes in the Property inspector without accessing the menus or panels that also control these attributes.

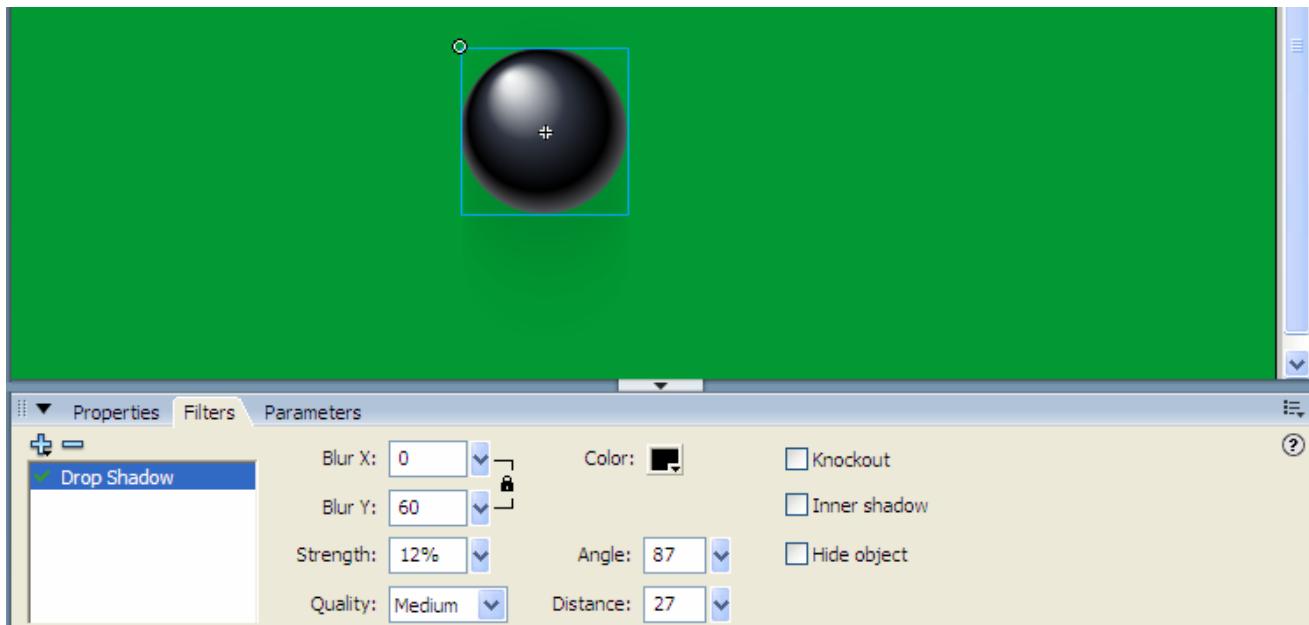
Depending on what is currently selected, the Property inspector displays information and settings for the current document, text, symbol, shape, bitmap, video, group, frame, or tool. When two or more different types of objects are selected, the Property inspector displays the total number of objects selected.

Open a Sample File and View the Property Inspector:

1. In Flash, select File > Open.
2. Use one of the following paths to browse to the completed file:
 - o In Windows, browse to *Unit_3\ AnimationandGradients* double-click *animation_and_gradients.fla*.

Test the movie. **Control > Test Movie**. You can see some of the blends and easing animation techniques that we will learn later.

3. After viewing the application, close the Flash Player window.
4. Click on the black ball on the Stage area. This is the 8ball movie clip
5. At the bottom of the Flash 8 interface locate the Property Inspector:



The Property inspector showing the properties for the MovieClip – 8ball.

6. Notice the new Filter Tab in the Property Inspector.
7. Have some fun, and modify the drop shadow settings.

We will learn more about Filter settings later on. Let's take a look at another new feature.

Viewing the Expanded Stage Area

Expanded Stage Work Area

You can use the area around the Stage to store graphics and other objects without having them appear on the Stage when you play the SWF file. Macromedia expanded this area, called the work area, to allow you to store more items there. Flash users often use the work area to store graphics they plan to animate on the Stage later, or to store objects that do not have a graphical representation during playback, such as data components.

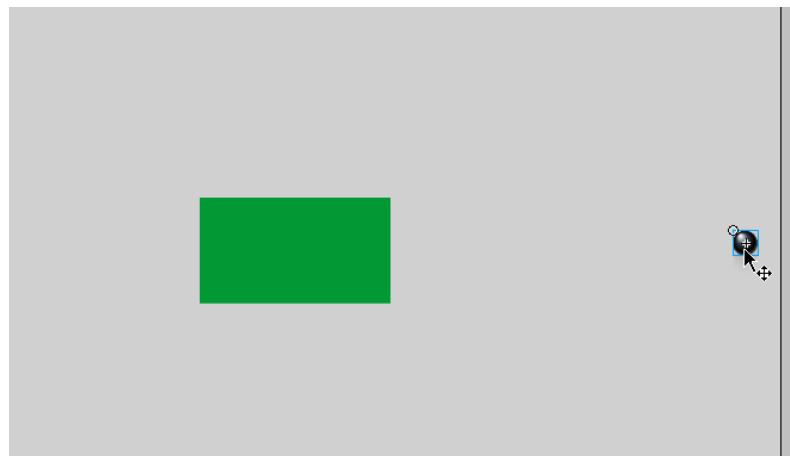
Animating item of the stage is easier than ever.

NOTE:

The only time the Stage will expand is when an object goes beyond the 2880 pixel limitation.

Viewing the Expanded Stage Area:

1. In Flash, select View > Work Area.
2. Using the **Zoom Out** tool , Zoom out al the way so you can see the entire grey work area.
3. With the **Selection** tool , click on the 8ball movie clip and drag it outside of the current grey workspace limit area.
4. Notice how the workspace now expands!



Move the 8ball movie clip outside of the current limit area.

Customizing the Flash Tools Panel

You can customize the Tools panel to specify which tools appear in the authoring environment. You use the Customize Tools panel dialog box to add or remove tools from the Tools panel.

You can display more than one tool in one location. When more than one tool is displayed in a location, the top tool in the group (the most recently used) is displayed with an arrow in the lower-right corner of its icon. When you press and hold the mouse button on the icon, the other tools in the group appear in a pop-up menu. You can then select a tool from the menu.

To customize the Tools panel:

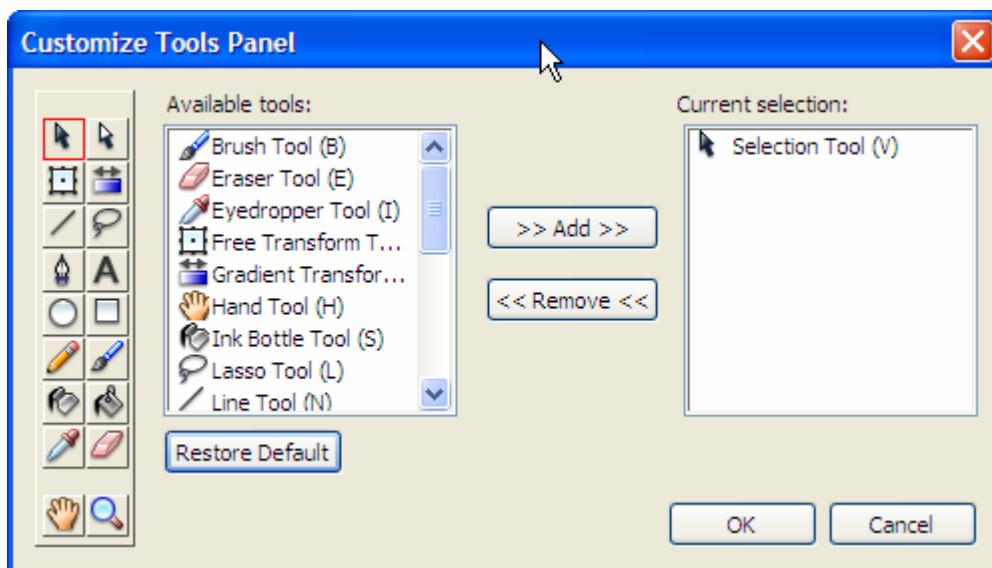
1. To display the Customize Tools panel dialog box, do one of the following:
 - (Windows) Select Edit > Customize Tools panel.
 - (Macintosh) Select Flash > Customize Tools panel.

The Available Tools menu indicates the tools that are currently available in the Flash. The Current Selection menu indicates the tool (or tools) currently assigned to the selected location in the Tools panel.

NOTE:

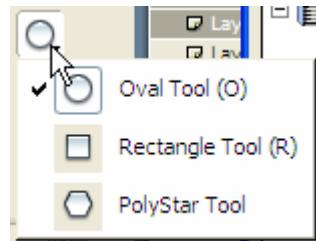
If more than one tool is assigned to one location in the Tools panel, a small arrow appears in the lower-right corner of the tool. This arrow indicates that additional tools are present in a pop-up menu. The same keyboard shortcut functions for all tools in the pop-up menu.

2. Click a tool in the Tools panel image or use the arrows to cycle through the tools to specify the location to which you want to assign another tool.



3. To add a tool to the selected location, select the tool in the Available Tools list and click the Add button. It is possible to assign a tool to more than one location.
4. To remove a tool from the selected location, select the tool in the Current Selection scroll list and click the Remove button.

5. Click OK to apply your changes and close the Customize Tools panel dialog box.
6. Notice now that when you have added tools, they appear as a drop down menu selection.



Here the Rectangle and PolyStar tool were added to the Oval Tool.

- Try to make a few more changes to the toolbar. You can even try the example above.

To restore the default Tools panel layout:

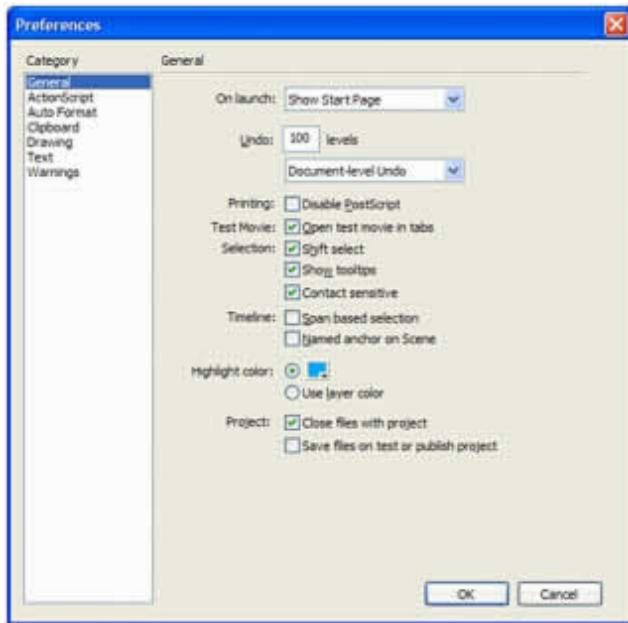
- Click Restore Default in the Customize Tools panel dialog box.

Changing Preferences

Improved Preferences dialog box

Macromedia streamlined the design of the Preferences dialog box and reorganized it for improved clarity and ease of use.

Flash lets you set preferences for general application operations, editing operations, and Clipboard operations.



The General category in the Preferences dialog box

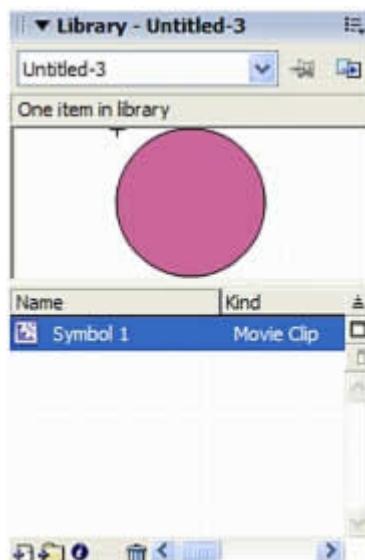
To set preferences:

1. Select **Edit > Preferences** (Windows) or **Flash > Preferences** (Macintosh).
2. In the Category list, select the one of the following:
 - o General
 - o ActionScript
 - o Auto Format
 - o Clipboard
 - o Drawing
 - o Text
 - o Warning

3. Select from the respective options to view the new way the panel is set up.
4. In the General section, under Project: choose the **Save files on test or publish project** checkbox.

Using the Flash 8 Library Panel

The Library panel is where you store and organize symbols created in Flash, as well as imported files, including bitmap graphics, sound files, and video clips. The Library panel lets you organize library items in folders, see how often an item is used in a document, and sort items by type.



The Library panel showing a movie clip symbol

To display the Library panel, do one of the following:

Select Window > Library.

Press Control+L (Windows) or Command+L (Macintosh).

Managing media assets with the library

The library in a Flash document stores media assets that you create or import for use in a Flash document. The library stores imported files such as video clips, sound clips, bitmaps, and imported vector artwork as well as symbols. A symbol is a graphic, button, or movie clip that you create once and can reuse multiple times. You can also create a font symbol. For information on symbols, see Using Symbols, Instances, and Library Assets and Creating font symbols.

The library also contains components that you have added to your document. Components appear in the library as compiled clips. For more information, see Components in the Library panel in Using Components.

The Library panel displays a scroll list with the names of all items in the library, which lets you view and organize these elements as you work. An icon next to an item's name in the Library panel indicates the item's file type. The Library panel has an options menu with commands for managing library items.

You can open the library of any Flash document while you are working in Flash, to make the library items from that file available for the current document.

You can create permanent libraries in your Flash application that is available whenever you start Flash. Flash also includes several sample libraries containing buttons, graphics, movie clips, and sounds that you can add to your Flash documents. The sample Flash libraries and permanent libraries that you create are listed in the **Window > Common Libraries** submenu. For more information, see [Working with common libraries](#).

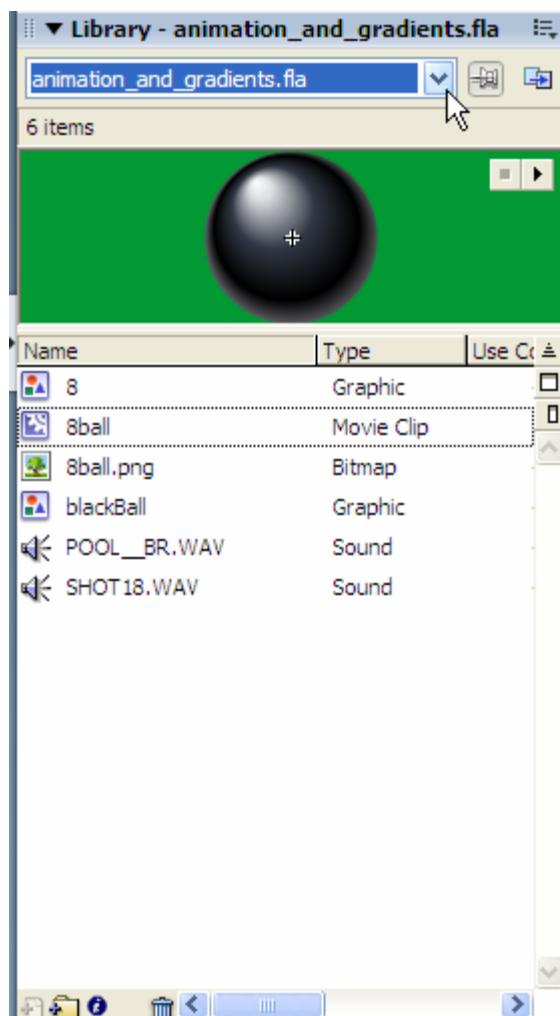
You can export library assets as a SWF file to a URL to create a runtime-shared library. This lets you link to the library assets from Flash documents that import symbols using runtime sharing. For more information, see [Using shared library assets](#).

Single Library panel

You can now use a single Library panel to view the library items of multiple Flash files simultaneously. You can switch back and forth as you edit multiple files.

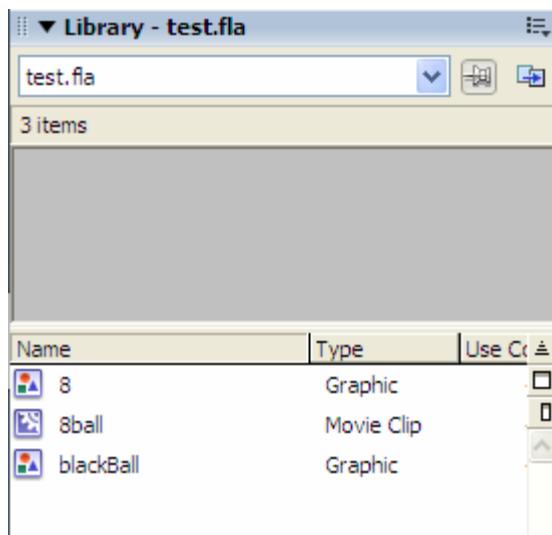
View the Single Library Function:

1. Select File > New, and make a blank Flash Document.
2. Save the file as **Test.fla** in the Unit 3 folder.



The new Library Panel in Flash 8.

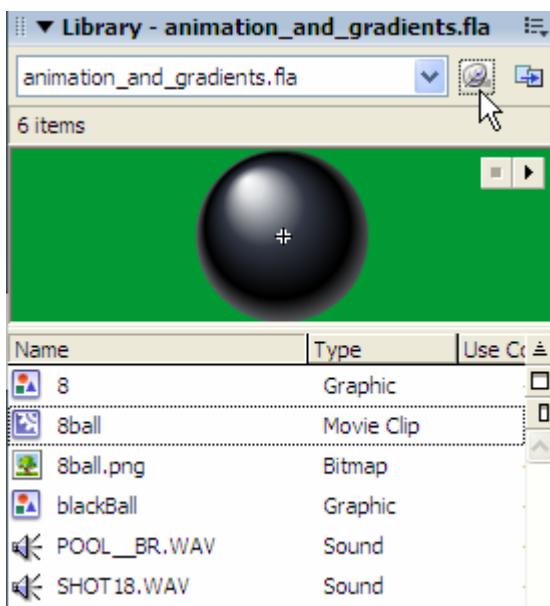
3. Click the drop down menu in the Library panel to toggle between the Libraries.
4. Copy and Paste a symbol into the **Test.fla** Library. In the **animation_and_gradients.fla** Library select the 8ball movie clip.
5. In Windows, **Right click** and select **Copy**.
6. Switch to the **Test.fla** Library and Right click again to **Paste** the symbol.



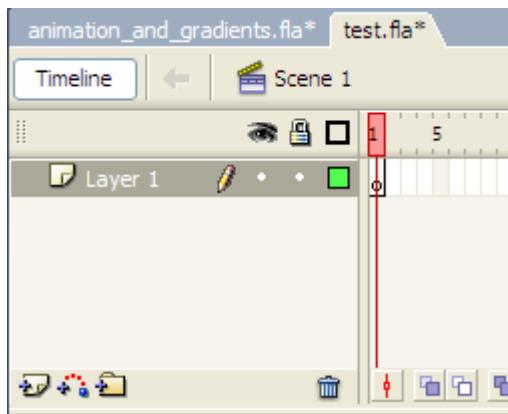
The 8ball movie clip has been copied to Test.fla. Notice it's associated Graphic symbols came along for the ride!

To Pin the Library Panel:

1. Switch back to the animation_and_gradients.fla Library using the drop down menu.
2. Click the Pin button  on the Library Panel.



3. Switch between your files by clicking on the file name tab.



Switch between files to see how the Library stays pinned

4. Notice how the same Library stays open.
5. Deselect the Pin, and switch between files. Notice how the Library Panel changes.

To resize the Library panel, do any of the following:

1. Drag the lower right corner of the panel.
2. Click the **Wide State**  button to enlarge the Library panel so it shows all the columns.
3. Click the **Narrow State**  button to reduce the width of the Library panel.
4. To change the width of columns:
5. Position the pointer between column headers and drag to resize.

NOTE: You cannot change the order of columns.

Using Tabbed Panels

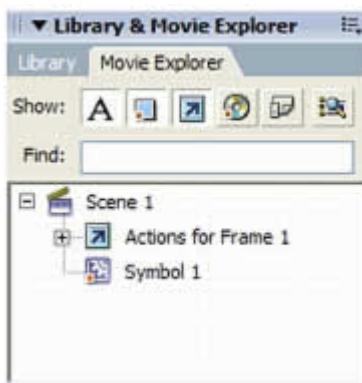
Improved Panel Management

Group panels together and navigate between panels with tabs, like other graphic applications. This is a old throwback that is a worthwhile addition to the Flash 8 environment.

To display multiple panels in a single panel window:

1. Click a panel's pop-up menu.
2. Select the *Group Panel Name With* option.
3. Select another panel to add the current panel to from the submenu.

The first panel is added as a tab to the second panel.



A tabbed panel showing the Library and Movie Explorer panels

Additional Interface Items

Drag and Drop Components to Library Panel

If you want to add a component to your Library, you can just drag it in the Library. There's no need to place one on the stage until you need it.

Macintosh document tabs

You can now open multiple Flash files in the same Flash application window and choose among them by using the document tabs at the top of the window.

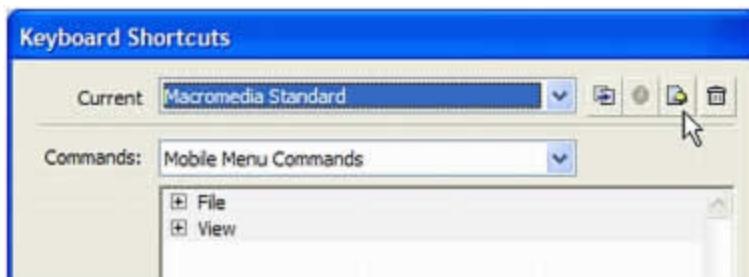
Export Keyboard Shortcuts as HTML

You can select keyboard shortcuts in Flash to match the shortcuts you use in other applications, or to streamline your Flash workflow. By default, Flash uses built-in keyboard shortcuts designed for the Flash application. You can also select a built-in keyboard shortcut set from one of several popular graphics applications, including Macromedia Fireworks, Adobe Illustrator, and Adobe Photoshop.

To create a custom keyboard shortcut set, you duplicate an existing set, and then add or remove shortcuts from the new set. You can also delete custom shortcut sets.

To view or print the current set of keyboard shortcuts:

1. Select **Edit > Keyboard Shortcuts**.
2. In the Keyboard Shortcuts dialog box, select the shortcut set you wish to view from the Current pop-up menu.
3. Click the Export Set as HTML button.



The Export Set as HTML button

4. In the Save As dialog box that appears, select a name and location for the exported HTML file. The default file name is the name of the selected shortcut set.
5. Click Save.
6. Find the exported file in the folder you selected and open the file in a web browser.
7. To print the file, use the browser's Print command.

Lab

No lab for this unit. Let's move on to Drawing!

New Features Summary

- Working with the Property Inspector – There is a new Filter Tab and other goodies.
- Viewing the Expanded Stage Area – Place objects as far out as you want!
- Customizing the Flash Toolbar – Modify your tools.
- Changing Preferences – New format, easier to read.
- Using the Flash 8 Library Panel – One Panel for all your files!
- Using Tabbed Panels – Organize for easier panel access.

Unit 4: Drawing Improvements

Unit Objectives:

After this unit, you will be able to:

- Understand and use the new Object drawing model
- Use the Gradient Tool
- Create new Gradients with ActionScript
- Use the new Shape tool Features
- Use the Free-Transform Tool
- Use Enhance Strokes

Unit Topics:

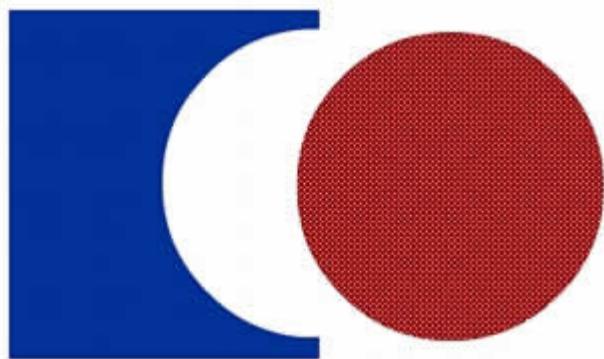
- The Object Drawing Model
- Using Gradients in Flash 8
- Building an 8-Ball with Gradients
- New Drawing Features in Flash 8

Object Drawing Model

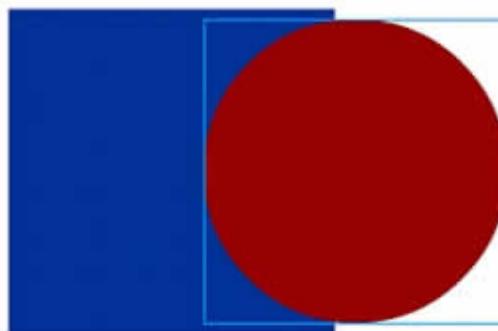
Previously in Flash, all shapes in the same layer on the Stage could affect the outlines of other overlapping shapes. You can now create shapes directly on the Stage that do not interfere with other shapes on the Stage. When you create a shape with the new Object Drawing model, the shape does not cause changes to other shapes that exist underneath the new shape.

Flash provides two drawing models that give you a great deal of flexibility when drawing shapes:

Merge Drawing model automatically merges shapes that you draw when you overlap them. If you select a shape that has been merged with another, and move it, the shape below it is permanently altered. For example, if you draw a square and overlay a circle on top of it, and then select the circle and move it, the portion of the square that overlaid the circle is removed.



Object Drawing model lets you draw shapes as separate objects that do not automatically merge together when overlaid. This lets you overlap shapes without altering their appearance should you choose to move them apart, or rearrange their appearance. Flash creates each shape as a separate object that you can individually manipulate. In previous versions of Flash, to overlap shapes without having their appearance altered, you had to draw each shape on its own layer.



When you select a shape created using the Object Drawing model, Flash surrounds the shape with a rectangular bounding box. You can use the Pointer tool to move the object by clicking the bounding box and dragging the shape anywhere you'd like to position it on the Stage.

NOTE:

You can set preferences for contact sensitivity when selecting shapes created using the Object Drawing model.

The new object drawing mode has to be enabled when using a drawing tool by pressing the "**J**" key or by pressing the "**Object Drawing button**" next to the magnet in the Tools panel.



Use the Object Drawing Model

1. Create a new Flash Document. **File > New**. Choose Flash Document.
2. Save the file as **Objects.fla** in the Unit 4 folder.
3. Draw any shapes that you want on top of each other.
4. Pull the shapes apart by using the Selection tool. Notice how the artwork is still destructive
5. Click on the Object Drawing button.
6. Draw a few more shapes on top of each other.
7. Try pulling them apart. They are now all seen as grouped objects!

Object-Level Undo Mode

You can now choose to keep track of the changes you make in Flash on a per-object basis. When you use this mode, each object on the Stage and in the library has its own undo list. This allows you to undo the changes you make to an object without having to undo changes to any other object.

Using the Undo, Redo, and Repeat menu commands

The Edit > Undo and Edit > Redo commands let you undo and redo steps as you work on Flash documents. The name of the command switches between Undo and Redo, depending on the action that was last performed.

Flash lets you specify either object-level or document-level Undo and Redo commands. This lets you undo or redo actions on individual objects, or all objects within the current document. The default behavior is document-level Undo and Redo.

You cannot undo the following actions when using object-level Undo:

- Enter Edit mode
- Exit edit mode
- Select unused library items
- Select library items
- Add library items
- Delete library items
- Duplicate library item
- Modify library item
- Modify library symbol behavior
- Rename library items
- Move library items
- Edit library items
- Import to library
- Create font symbols
- Create library folders
- Expand all library folders
- Create video stream symbols
- Convert to compiled clip
- JSFL library edits
- Modify bitmap symbol properties
- Modify sound symbol properties
- Modify library item linkage
- Convert to symbol
- Create new symbol
- Run JSFL command
- Run JSFL file
- Modify movie properties
- Import
- Create scene

- Delete scene
- Duplicate scene
- Rename scene
- Move scene

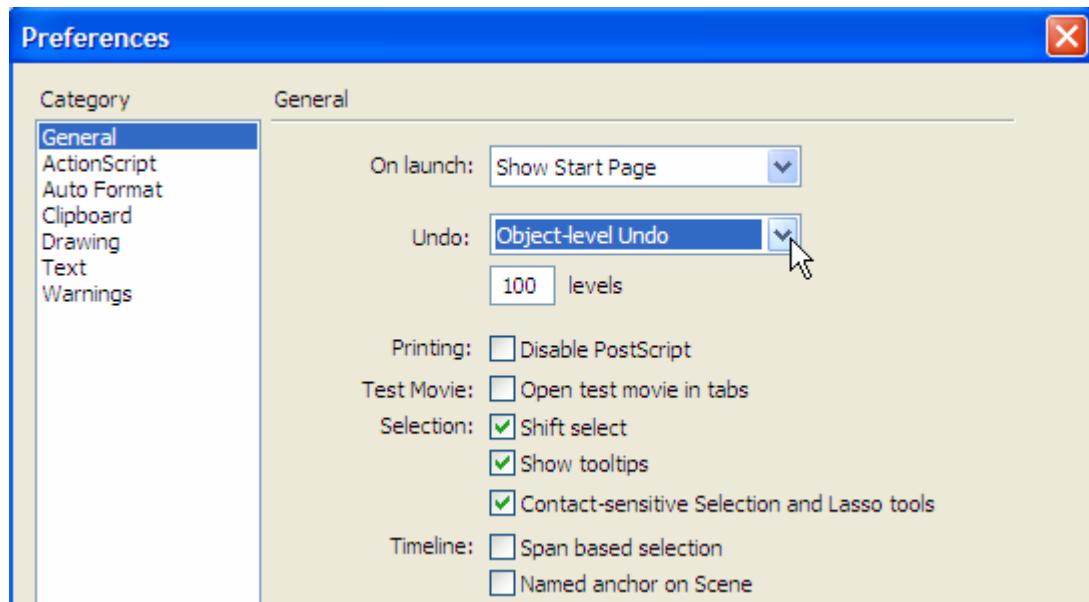
To remove deleted items from a document after using the Undo command, you use the **Save and Compact** command. See Saving documents when you undo steps.

You can use the Repeat command to reapply a step to the same object or to a different object. For example, if you move a shape named shape_A, you can select **Edit > Repeat** to move the shape again, or you can select another shape, shape_B, and select **Edit > Repeat** to move the second shape by the same amount.

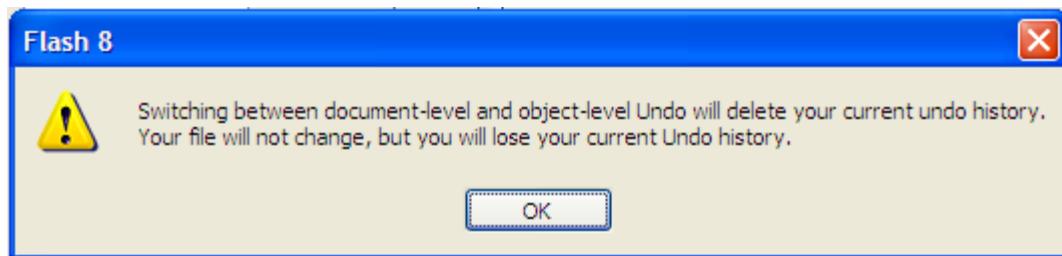
By default, Flash supports 100 levels of undo for the Undo menu command. You can select the number of undo and redo levels, from 2 to 9999, in Flash Preferences.

Set Up Your Undo Editing Preference

1. Select **Edit > Preferences**. General Category.



2. Click the Undo Dropdown and change it to **Object Level Undo**.
3. You will see the following dialog box:



4. Click **OK**.

Transforming gradient and bitmap fills

You can transform a gradient or bitmap fill by adjusting the size, direction, or center of the fill. To transform a gradient or bitmap fill, you use the Gradient Transform tool. One of the new features of Flash 8 is adjusting the Focal Point of a gradient.

To adjust a gradient or bitmap fill with the Gradient Transform tool:

1. Create a new Flash Document (File > New), and draw a shape with a radial gradient fill.

Select the Gradient Transform tool. 

2. Click an area filled with a gradient or bitmap fill. A bounding box with editing handles is displayed. When the pointer is over any one of these handles, it changes to indicate the function of the handle.

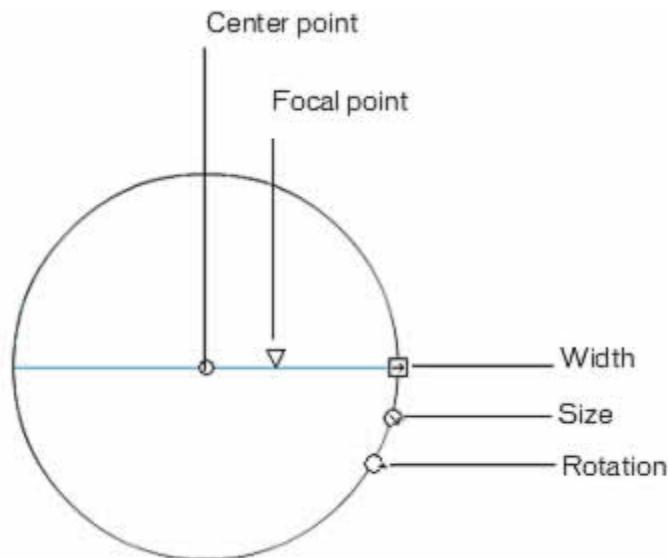
Center point Select and move the center point handle to change the center point of the gradient. The rollover icon for the center point handle is a four-way arrow.

Focal point Select the focal point handle to change the focal point of a radial gradient. The focal point handle is displayed only when you select a radial gradient; the rollover icon for the focal point handle is an inverted triangle.

Size Click and move the middle handle icon on the edge of the bounding box to adjust the size of the gradient. The rollover icon for the size handle is a circle with an arrow inside of it.

Rotation Click and move the bottom handle on the edge of the bounding box to adjust the rotation of the gradient. The rollover icon for the rotation handle is four arrows in the shape of a circle.

Width Click and move the square handle to adjust the width of the gradient. The rollover icon for the width handle is a double-ended arrow.



Press Shift to constrain the direction of a linear gradient fill to multiples of 45°.

Reshape the gradient or fill in any of the following ways:

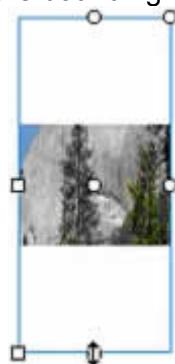
To reposition the center point of the gradient or bitmap fill, drag the center point.



To change the width of the gradient or bitmap fill, drag the square handle on the side of the bounding box. (This option resizes only the fill, not the object containing the fill.)



To change the height of the gradient or bitmap fill, drag the square handle at the bottom of the bounding box.



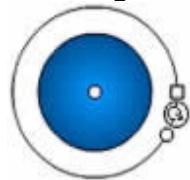
To rotate the gradient or bitmap fill, drag the circular rotation handle at the corner. You can also drag the lowest handle on the bounding circle of a circular gradient or fill.



To scale a linear gradient or a fill, drag the square handle at the center of the bounding box.



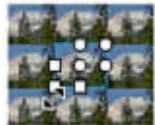
To change the focal point of a circular gradient, drag the middle circular handle on the bounding circle.



To skew or slant a fill within a shape, drag one of the circular handles on the top or right side of the bounding box.



To tile a bitmap inside a shape, scale the fill.



New Gradient Color Options

The Color Mixer provides options for changing the color of strokes and fills, as well as creating multicolor gradients. You can use gradients to produce a wide range of effects, such as giving an illusion of depth to a two-dimensional object. For example, you can use a gradient to transform a simple, two-dimensional circle into a sphere, with the illusion of light illuminating the surface at one angle and casting a shadow to the opposite side of the sphere.



The Color Mixer with the gradient controls displayed.

The Color Mixer has the following options:

Stroke color changes the color of the stroke, or the border, of a graphic object.

Fill Color changes the color of the fill. The fill is the area of color that fills up the shape.

Type pop-up menu changes the fill style:

- **None** removes the fill.
- **Solid** provides a solid, single fill color.
- **Linear** produces a gradient that blends on a linear path.
- **Radial** produces a gradient that blends outward in a circular path from a central focal point.
- **Bitmap** lets you tile the selected fill area with a bitmap image that you can select. When you choose Bitmap, a dialog box lets you select a bitmap image on your local computer, and add it to the library. You can apply this bitmap as a fill; the appearance is something similar to a mosaic tile pattern with the image repeated within the shape.

RGB let you change the density of the red, blue, and green (RGB) colors in a fill.

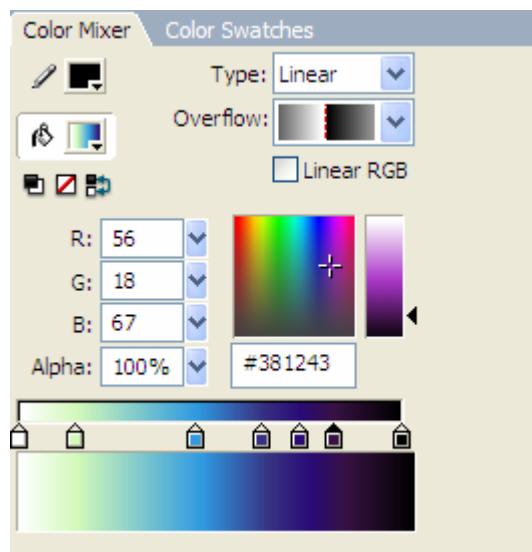
Alpha sets the opacity for a solid fill, or the currently selected slider for a gradient fill. An alpha value of 0% creates an invisible (or transparent) fill; an alpha value of 100% creates an opaque fill.

Current Color Swatch displays the currently selected color. If you select a gradient fill type (Linear or Radial) from the fill Type pop-up menu, the Current Color Swatch displays the color transitions within the gradient you create.

Color Picker lets you select a color visually. Click the Color Picker and drag the cross-hair pointer around until you find the color you want.

Hexadecimal value displays the current color's hexadecimal value. To change the color using the hexadecimal value, type in a new value. Hexadecimal color values (also called hex values) are 6-digit alphanumeric combinations that represent a color.

Brand New in Flash 8



Now you can apply up to 16 colors!

Overflow lets you control colors applied past the limits of a linear or radial gradient. The overflow modes are extend (the default mode), reflect, and repeat.

- **Extend** applies the colors you specify past the end of the gradient.
- **Reflect** causes the gradient colors to fill the shape using a reflective mirroring effect. The gradients you specify are repeated in a pattern from the beginning of the gradient to the end, and then repeated in the opposite sequence from the end of the gradient to the beginning, and then back to the beginning of the gradient to the end until the selected shape is filled.
- **Repeat** repeats the gradient from the beginning of the gradient to the end until the selected shape is filled.

Linear RGB creates an SVG-compliant (Scalable Vector Graphics) linear or radial gradient.

Building the 8-Ball with Gradients

As you examine the finished version of an application you'll create, you will also look at the Flash workspace.

In this section, you will complete the following tasks:

- Open the authoring document
- Review the completed FLA file
- Close the completed FLA file

Open the authoring document

It's helpful to analyze the completed authoring document, which is a FLA file, to see how the author designed the illustration and understand what you are going to create.

On most computers, you will find the Gradients tutorial folder in the following locations:

- In Windows: ***Unit_4 folder/Gradients/gradients_finish.fla***

Double-click the file to open it in Flash. You now see the completed tutorial file in the Flash authoring environment.

Review the completed FLA file

In the completed FLA file, you will see the combined effects of several gradients. The illustration looks like this:



There are five gradients in the illustration:

- A gradient from black to green on the background.
- A gradient from black to green to black on the bottom of the eight ball.
- Another from white to black in the highlight on the top of the eight ball.
- A subtle gradient on the white circle surrounding the number "8."
- A radial gradient in the shadow beneath the eight ball.

Close the completed FLA file

Open the starter document

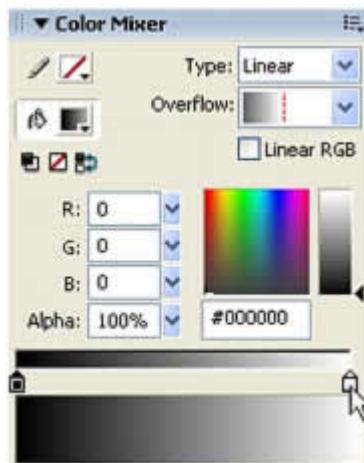
Now that you have seen the completed file, it is time to create your own Flash document. To get started, you'll open a starter file that contains a few graphics to which you will apply gradients.

1. In Flash, select File > Open.
2. Navigate to the following directory:
 - In Windows: ***Unit_4 folder/Gradients/gradients_start.fla***
3. Open the **gradients_start.fla** file.

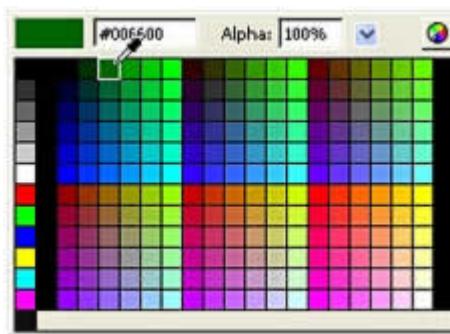
Apply a linear gradient

In this section you will add a linear gradient to the background.

1. Click the Selection tool in the Tools panel.
2. Select the gray box in the layer named Background.
3. Select Window > Color Mixer to display the Color Mixer panel.
4. In the Color Mixer's Type pop-up menu, select Linear.
5. Double-click the right gradient color swatch and select the color green (#006600).



Selecting the right gradient color swatch in the Color Mixer



Selecting the color green #006600 in the Color Picker

6. Double-click the left gradient color swatch and select the color black (#000000).
7. Select the Gradient Transform tool from the Tools panel. The Gradient Transform controls appear on the Stage around the gradient.

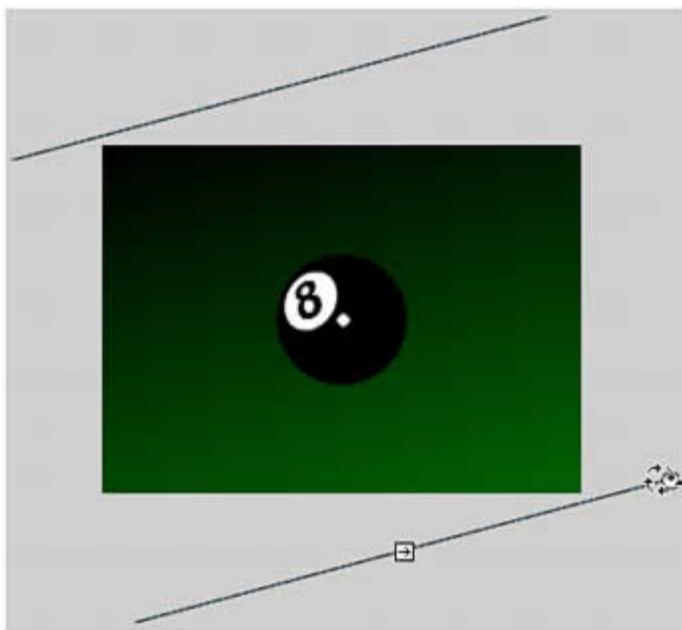


The Gradient Transform tool

8. Drag the Gradient Rotate handle to rotate the linear gradient clockwise as shown.

The Gradient Rotate handle





Rotating the gradient clockwise

9. Lock the Background layer in the Timeline to prevent further changes to this layer.
10. Select **File > Save** to save your FLA file.

Create a Radial Gradient

Next, you will add a radial gradient to the black eight ball.

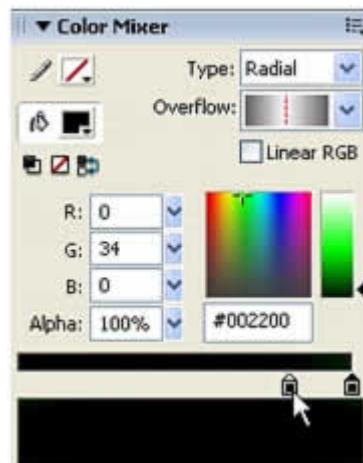
1. Double-click the black circle in the layer called Ball. This opens the group containing the eight-ball shape.
2. Select the black circle shape. You will apply a gradient to this shape. Do not select the number "8" on the eight ball.
3. In the Color Mixer panel select Radial from the Type pop-up menu. Select Mirror Overflow mode from the Overflow pop-up menu.



The correct Radial Type and Mirror Overflow settings

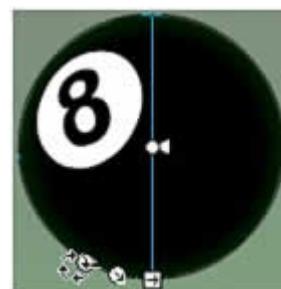
4. Double-click the left gradient color swatch and select the color black (#000000).
5. Double-click the right gradient color swatch and type **002200** into the color text box. Press Enter.

6. Drag the left gradient color swatch to the right about three-fourths of the way as shown in the following illustration. This makes the green part of the gradient appear only in the **outer 25%** of the ball shape.



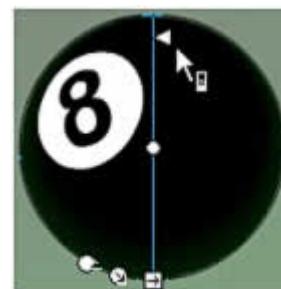
Dragging a gradient color swatch

7. Select the **Zoom tool** from the Tools panel and click the circle shape to magnify it.
8. Select the **Gradient Transform tool** in the Tools panel.
9. Rotate the radial gradient **90°** clockwise by dragging the Gradient Rotate handle.



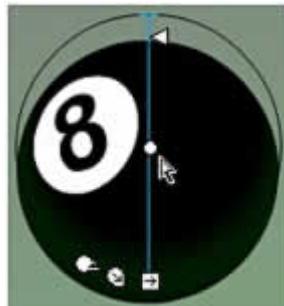
Dragging to rotate the gradient 90° clockwise.

10. Select the **Focal Point control** and drag it near the top of the circle.



Dragging the Focal Point control toward the top of the circle shape

11. Select the center control point and drag the entire gradient upward a short distance as shown in the following illustration. The mirrored overflow gradient is at the bottom of the circle.



Dragging the center control point upwards

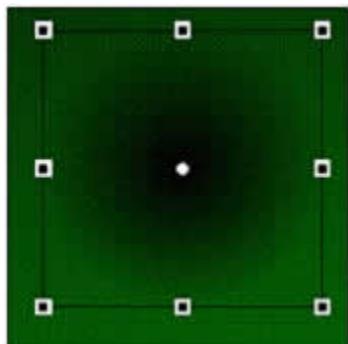
12. Select the **Selection tool** in the Tools panel.
13. Double-click the **Zoom tool** to return the Stage area to a view of 100%.
14. Double-click the Stage area to deselect the eight-ball group.
15. Select **File > Save** to save your FLA file.

Create a Transform Gradient with a Shape

In this section, you will create a gradient and then use the Free Transform tool to change its shape.

1. Select the **Shadow layer** in the Timeline.
2. Select the **Oval tool** in the Tools panel.
3. Hold down Shift and then drag on the Stage with the Oval tool to draw a circle approximately **150 pixels** square. This should be about the same diameter as the eight ball.
4. In the Color Mixer's Type menu, select **Radial**.
5. Move the left gradient color swatch all the way to the left. You should have one swatch all the way to the left and one all the way to the right.
6. Double-click the left swatch and select the color **black (#000000)** in the Color Picker.
7. Double-click the right swatch to display the Color Picker.
8. Select the color **black (#000000)**.
9. In the Color Picker, drag the Alpha slider down to zero. This creates a gradient from black to transparent, allowing the green background to show through the edge of the circle you just drew.
10. Select the **Gradient Transform tool** from the Tools panel and drag the Focal Point control of the gradient back to the center of the circle.

11. Select the **Free Transform tool** and scale the circle along the *y* (vertical) axis by dragging the top-center handle downward as shown in the following illustration.



The gradient is transformed along with the shape transformation.

12. Select the **Selection tool** from the Tools panel.
13. Drag the shadow shape under the eight ball with the Selection tool.
14. Click outside the Stage to deselect the shadow.
15. Select **File > Save** to save your FLA file.

Apply the Finishing Touches

To complete the illustration, you will apply two more linear gradients: one to the white circle on the eight ball and one to make a highlight on the top of the eight ball.

Apply a gradient to the white circle on the eight ball

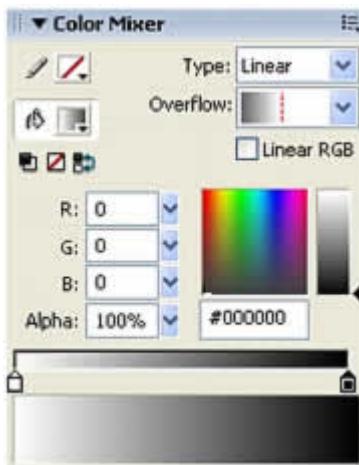
Next you will add a gradient to the white circle on the eight ball graphic.

1. Select the **Selection tool** from the Tools panel.
2. Double-click the eight ball group to edit the group.
3. Double-click the white circle group.
4. Select the white circle shape.
5. In the Color Mixer panel, assign the white circle shape a linear gradient by selecting **Linear** from the Type menu.
6. Select **Nonrepeating** from the Overflow menu.
7. Drag the left gradient color swatch all the way to the left and double-click it.
8. Select the color **white (#FFFFFF)** from the Color Picker.
9. Set the Alpha value for the white color swatch to **100%**.
10. Drag the right color swatch all the way to the right and double-click it.

11. Select the color **black (#000000)** from the Color Picker.

12. Set the Alpha value for the black color swatch to **100%**.

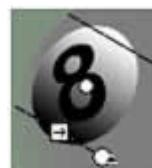
Your gradient settings should match those in the following illustration:



The Color Mixer with the correct settings for the gradient on the white circle

13. Select the Gradient Transform tool from the Tools panel.

14. Drag the Gradient Rotate handle approximately **120°** clockwise.



The gradient rotated 120°

15. Select the Selection tool in the Tools panel.

16. Double-click the Stage twice to exit the white circle group and the eight-ball group.

17. Select **File > Save** to save your FLA file.

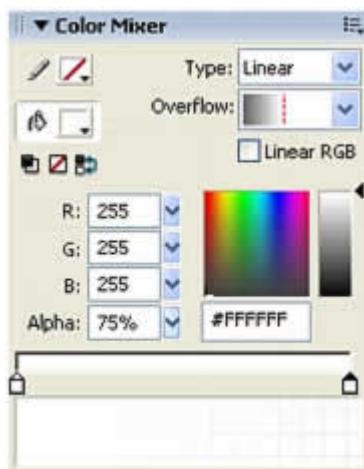
Apply a linear gradient to the highlight

In this section you will apply a linear gradient.

1. Click the red X icon in the layer named **Highlight** to reveal the shape in that layer. The shape appears at the top of the eight ball.
2. With the **Selection tool**, double-click the highlight shape to edit its group.
3. Select the highlight shape.
4. In the **Color Mixer**, select **Linear** from the Type menu.

5. Select **Non-repeating** from the Overflow menu.
6. Drag the left gradient color swatch all the way to the left and double-click it.
7. Select the color **white (#FFFFFF)** from the Color Picker.
8. Set the Alpha value for the left swatch to 0%.
9. Drag the right gradient color swatch all the way to the right and double-click it.
10. Select the color white from the Color Picker.
11. Set the Alpha value for the right swatch to **75%**.

Your gradient settings should match those in the following illustration:



The Color Mixer with the correct settings for the gradient on the highlight shape

12. With the highlight shape still selected, select the Gradient Transform tool from the Tools panel.
13. Drag the Gradient Rotate control **90°** counterclockwise.
14. Drag the Gradient Scale control downward until it touches the top of the highlight shape.



The Gradient Scale control



Dragging the Gradient Scale control downward to touch the highlight shape

15. Select the Selection tool from the Tools panel.

16. Double-click outside the Stage twice to deselect the highlight group.

The finished Flash illustration looks like this:



17. Select **File > Save** to save your FLA file.

To close the document, select **File > Close**.

Using Complex Gradient Fills with ActionScript

The Flash Drawing API supports gradient fills as well as solid fills. The following procedure creates a new movie clip on the Stage, use the Drawing API to create a square, and then fills the square with a radial red and blue gradient.

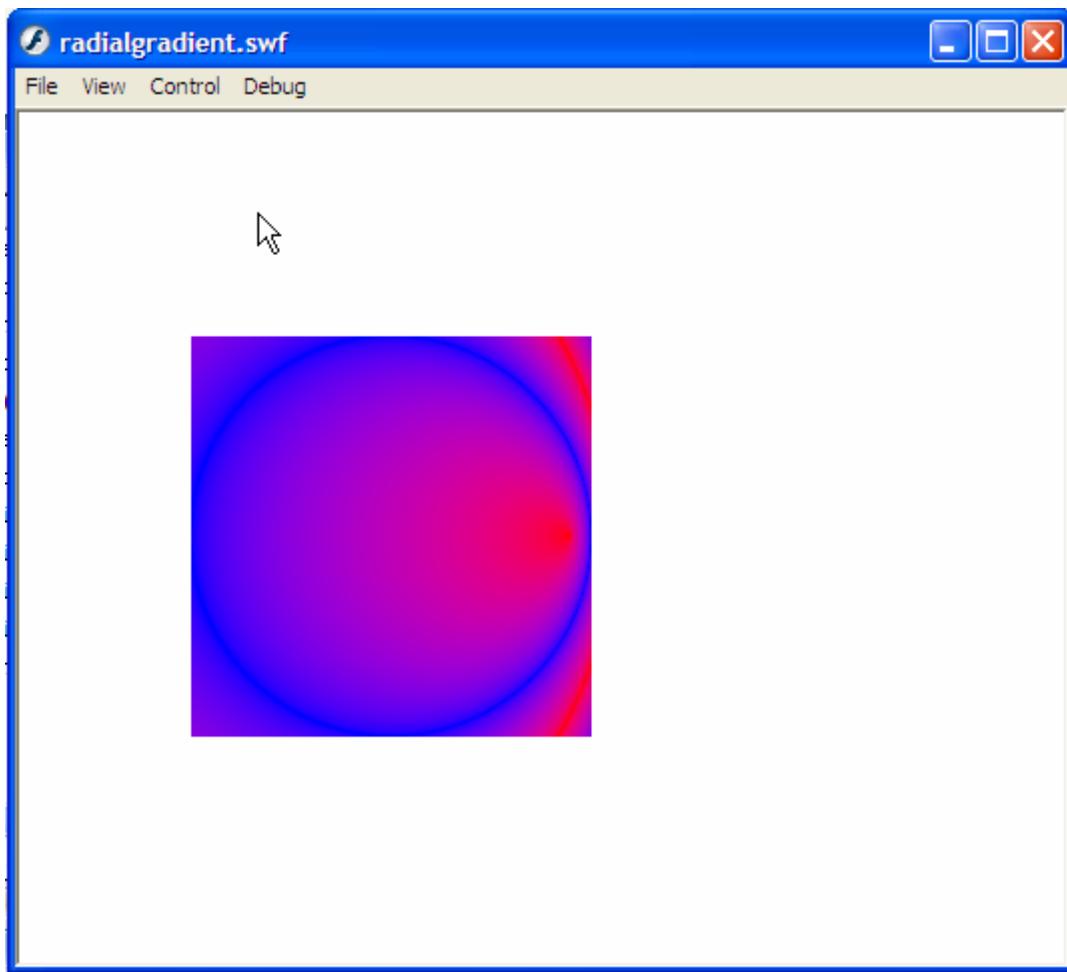
To create a complex gradient:

1. Create a new Flash document and save it as **radialgradient.fla** in the Unit 4 folder.
2. Add the following ActionScript to Frame 1 of the Timeline:

```
this.createEmptyMovieClip("gradient_mc", 10);
var fillType:String = "radial";
var colors:Array = [0xFF0000, 0x0000FF];
var alphas:Array = [100, 100];
var ratios:Array = [0, 0xFF];
var matrix:Object = {a:200, b:0, c:0, d:0, e:200, f:0, g:200,
h:200, i:1};
var spreadMethod:String = "reflect";
var interpolationMethod:String = "linearRGB";
var focalPointRatio:Number = 0.9;
with (gradient_mc) {
    beginGradientFill(fillType, colors, alphas, ratios, matrix,
spreadMethod, interpolationMethod, focalPointRatio);
    moveTo(100, 100);
    lineTo(100, 300);
    lineTo(300, 300);
    lineTo(300, 100);
    lineTo(100, 100);
    endFill();
}
```

The preceding ActionScript code uses the Drawing API to create a square on the Stage and calls the `beginGradientFill()` method to fill the square with a red and blue circular gradient.

3. Save the Flash document and select **Control > Test Movie** to view the Flash file.



Using Script Assist with Gradients

Script Assist mode

A new assisted mode in the Actions panel allows you to create scripts without detailed knowledge of ActionScript.

For users who are new to ActionScript, or for those who want to add simple interactivity without having to learn the ActionScript language and its syntax, you can opt to use Script Assist to help you more easily add ActionScript to your Flash documents.

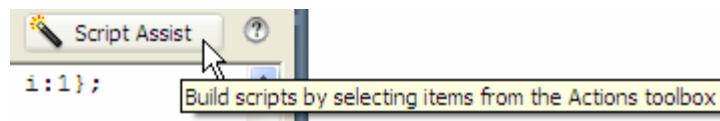
NOTE:

Script Assist is intended to help you format your scripts in such a way as to avoid the syntax and logic errors a novice user might encounter when writing ActionScript. However, to use Script Assist you must become familiar with ActionScript, and know what methods, functions, and variables to use when creating your scripts.

Using Script Assist to view ActionScript

To add an action to a Flash document, you must attach it to a button or movie clip, or to a frame in the Timeline. The Actions panel lets you select, drag and drop, rearrange, and delete actions.

1. To write ActionScript using Script Assist:
1. Select Window > Actions.
2. The Actions panel appears.
3. Click the Script Assist button.

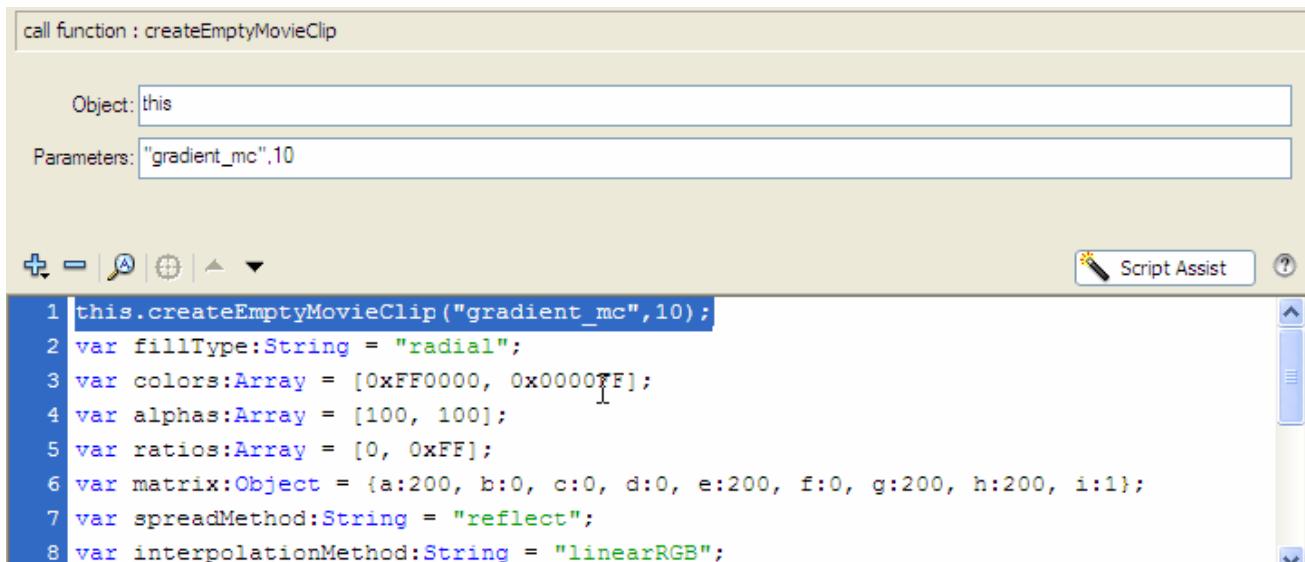


4. The Actions panel enters Script Assist mode.

NOTE:

If the Actions panel contains ActionScript code when you click the Script Assist button, Flash compiles the existing code. If there are errors in the code, you will not be able to use Script Assist until you fix the current code selection.

5. Click on each line of code. Notice how Script Assist gives you the details.



Script Assist giving specific information on each line of code.

When Script Assist is enabled, the Actions panel user interface and behaviors change in the following ways:

- **The Add (+) button** functions differently in Script Assist mode. When the focus is on the ActionScript window, it adds the selection after the currently selected text block. If the focus is in the edit pane, the selection is added to that text block.

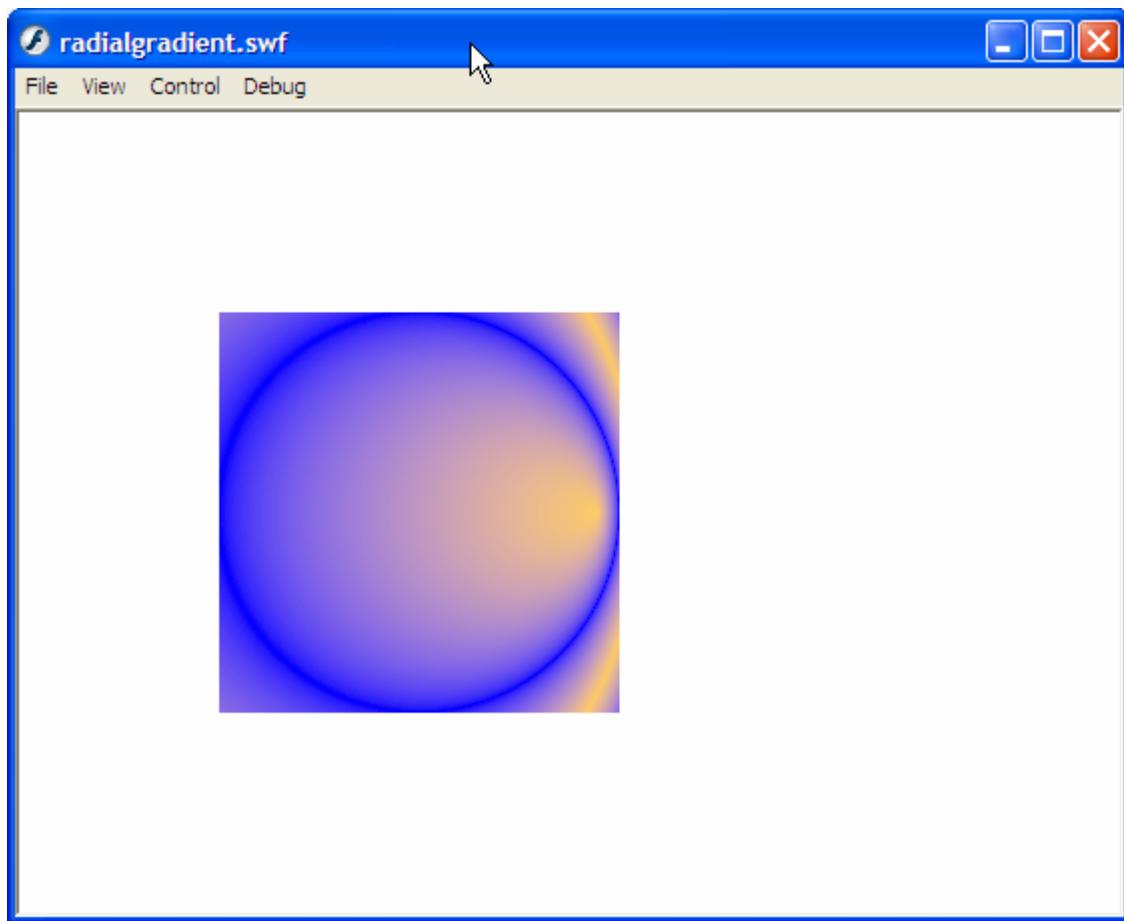
- **The Remove (-)** button lets you remove the current selection in the scrolling text area.
- **The Up and Down Arrow** buttons let you move the current selection in the scrolling text area forward or backward within the code.
- **The Check Syntax, Auto Format, Show Code Hint, and Debug Options** buttons and menu items normally visible in the Actions panel are disabled, as they do not apply to Script Assist mode.
- **The Insert Target** button is disabled unless you are editing a field. Clicking Insert Target places the resulting code in the current edit field.
- To view a description of an action, do one of the following:
 - Click a category in the Actions toolbox to display the actions in that category, and click an action.
 - Select a line of code in the Script pane.
 - The description appears at the upper left of the Actions panel, beginning with the Property or Event name.

Change the Values of the Gradient using Script Assist

1. Click on the following line of code:

```
var colors:Array = [0xFF0000, 0x0000FF];
```

1. Change the first color value to **0xFFCC66**. Don't delete the brackets!
2. Test your movie. **Control > Test Movie**.



You gradient color changes.

New Drawing Items In Flash 8

Transforming objects freely

You can use the Free Transform tool to freely transform objects, groups, instances, or text blocks. You can perform individual transformations or combine several transformations, such as moving, rotating, scaling, skewing, and distortion.

To transform freely:

1. Select a graphic object, group, instance, or text block on the Stage. No Movie clips!
2.  Click the Free Transform tool.

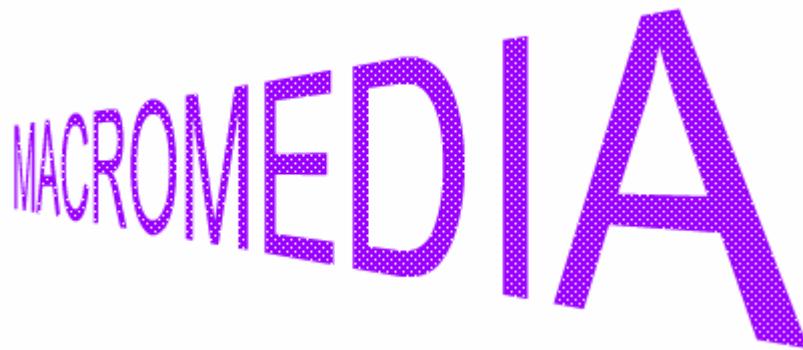
Moving the pointer over and around the selection changes the pointer to indicate which transformation function is available.

3. Drag the handles to transform the selection, as follows:
 - **To move the selection**, position the pointer over the object within the bounding box, and drag the object to a new position. Do not drag the transformation point.
 - **To set the center of rotation** or scaling, drag the transformation point to a new location.
 - **To rotate the selection**, position the pointer just outside a corner handle and drag. The selection rotates around the transformation point.
Shift-drag to rotate in 45° increments.
Alt-drag (Windows) or Option-drag (Macintosh) to rotate around the opposite corner.
 - **To scale the selection**, drag a corner handle diagonally to scale in two dimensions.
To scale in the respective direction only, drag a corner handle or a side handle horizontally or vertically.
Shift-drag to resize proportionally.
 - **To skew the selection**, position the pointer on the outline between the transformation handles and drag.
 - **To distort shapes**, press Control (Windows) or Command (Macintosh) and drag a corner handle or a side handle.
 - Shift-Control-click to drag (Windows) or Shift-Command-click to drag (Macintosh) a corner handle to **taper the object**--to move the selected corner and the adjoining corner equal distances from their origins.

NOTE:

The Free Transform tool cannot transform symbols, bitmaps, video objects, sounds, gradients, or text. If a multiple selection contains any of these, only the shape objects are distorted. To transform a text block, first convert the characters to shape objects.

4. To end the transformation, click outside the selected object, group, instance, or text block.



Now you can taper shapes. This text was broken apart before transforming.

Distorting Objects

When you apply a Distort transformation to a selected object, dragging a corner handle or an edge handle on the bounding box moves the corner or edge and realigns the adjoining edges. Shift-drag a corner point to *taper* the object—that is, move that corner and the adjoining corner an equal distance and in the opposite direction from each other. The adjoining corner is the corner on the same axis as the direction you drag. Control-click to drag (Windows) or Command-click to drag a middle point on an edge to move the entire edge freely.

You can distort graphic objects by using the Distort command. You can also use the tool options on the tool panel:



To distort graphic objects:

1. Select a graphic object or objects on the Stage.
2. Select Modify > Transform > Distort.
3. Place the pointer on one of the transformation handles and drag.
4. To end the transformation, click outside the selected object or objects.

Modifying Shapes with the Envelope Modifier

The Envelope modifier lets you warp and distort objects. An envelope is a bounding box that contains one or more objects. Changes made to an envelope's shape affect the shape of the objects contained within the envelope. You edit the shape of an envelope by adjusting its points and tangent handles.

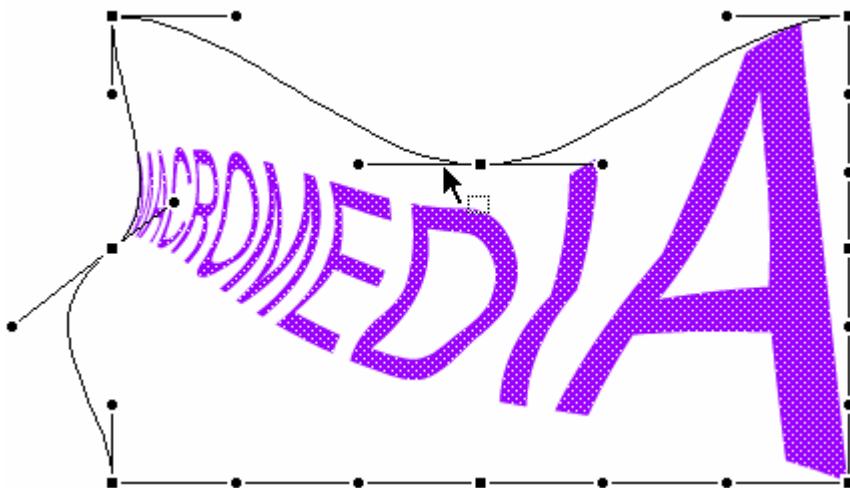
To modify a shape with the Envelope modifier:

1. Select a shape on the Stage.

NOTE:

The Envelope modifier cannot modify symbols, bitmaps, video objects, sounds, gradients, object groups, or text. If a multiple selection contains any of these, only the shape objects are distorted. To modify text, first convert the characters to shape objects.

2. Select Modify > Transform > Envelope.
3. Drag the points and tangent handles to modify the envelope.



The Envelope in action!

Drawing straight lines, ovals, and rectangles

You can use the Line, Oval, and Rectangle tools to easily create these basic geometric shapes. The Oval and Rectangle tools create stroked and filled shapes. The Rectangle tool lets you create rectangles with square or rounded corners.

To draw a straight line, oval, or rectangle:

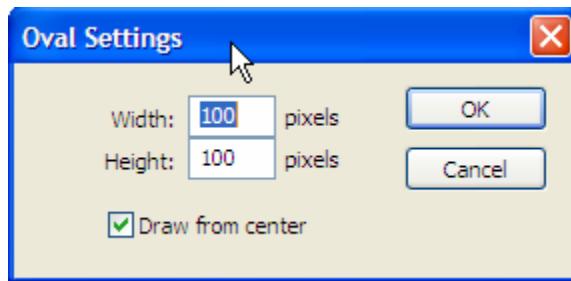
1. Select the Line, Oval, or Rectangle tool.
2. Select Window > Properties and select stroke and fill attributes in the Property inspector.
3. For the Rectangle tool, specify rounded corners by clicking the Round Rectangle modifier and entering a corner radius value. A value of zero creates square corners.
4. Drag on the Stage. If you are using the Rectangle tool, press the Up Arrow and Down Arrow keys while dragging to adjust the radius of rounded corners.

For the Oval and Rectangle tools, Shift-drag to constrain the shapes to circles and squares.

New in Flash 8

To specify a specific size of Oval or Rectangle in pixels, press the Alt key (Windows) or Option key (Macintosh) with the Oval or Rectangle tool selected, and click the Stage to display the Oval and Rectangle Settings dialog box.

- For ovals, you can specify the width and height in pixels, and whether to draw the oval from the center.
- For rectangles, you can specify the width and height in pixels, the radius of the rectangle corners for rounded corners, and whether to draw the rectangle from the center.



For the Line tool, Shift-drag to constrain the angle of lines to multiples of 45°.

Using the Stroke Color and Fill Color controls in the Property inspector

To change the stroke color, style, and weight for a selected object, you can use the Stroke Color controls in the Property inspector. For stroke style, you can choose from styles that are preloaded with Flash, or you can create a custom style.

To select a solid color fill, you can use the Fill Color control in the Property inspector.

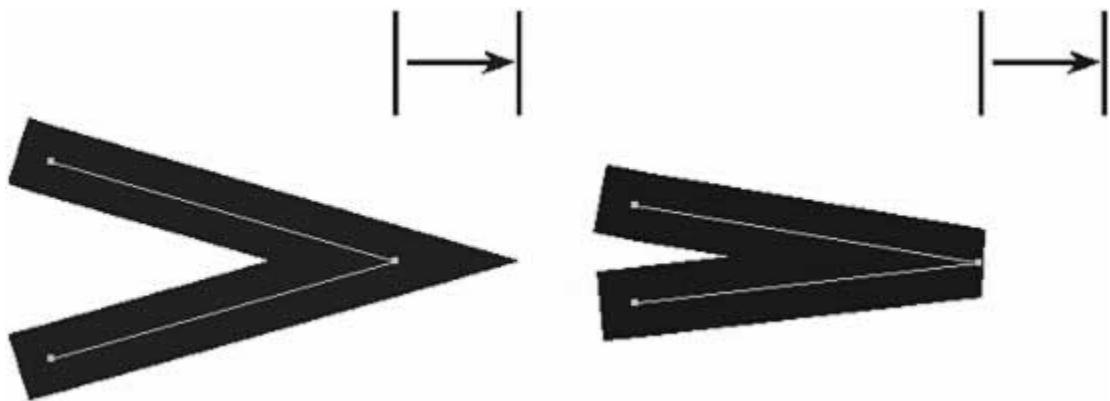
To select a stroke color, style, and weight using the Property inspector:

1. Select an object or objects on the Stage or draw a new shape.
2. To select a stroke style, click the triangle next to the Style pop-up menu and select an option from the menu. To create a custom style, select Custom from the Property inspector, and then select options in the Stroke Style dialog box and click OK. Also, try to give the stroke a gradient fill.
3. To select a stroke weight, click the triangle next to the Weight pop-up menu and set the slider at the weight you want.
4. Specify a stroke height by doing one of the following:
 - Select one of the preset values from the height pop-up menu. Preset values appear in points.
 - **New Flash 8 Feature:** Type a value from 0 to 200 in the height text box, and press Enter.
5. **New Flash 8 Feature:** Select the Stroke Hinting check box to enable stroke hinting. Stroke hinting adjusts line and curve anchors on full pixels, preventing blurry vertical or horizontal lines.
6. **New Flash 8 Feature :**Select a Cap option to set the style for a path end:
 - **None** is flush with the path's end
 - **Round**
 - **Square** extends beyond the path by half the stroke width
7. Select a Join option to define how two path segments meet: Miter, Round, or Bevel. To change the corners in an open or closed path, select a path and select another join option.



8. To avoid beveling a Miter join, enter a Miter limit.

Line lengths exceeding this value are squared off instead of pointed. For example, a Miter limit of 2 for a 3-point stroke means that when the length of the point is twice the stroke weight, Flash removes the limit point.



Lab

- Draw some text and distort it using the Envelope tool. You will need to Break the text apart. Remember – no objects when distorting!

New Features Summary

- Object Drawing Mode
- Object Level Undo
- 16 Colors in Gradients
- Modify Gradients
- Program Gradients with ActionScript
- New Script Assist Tool
- Free Distort and Envelope
- Caps,Joins and Miter on Strokes

Unit 5: Expressiveness Features and Animation

Unit Objectives:

After this unit, you will be able to:

- Use Blend Modes and Filters
- Use ActionScript to Control Expressiveness Features
- Import Blends and Filters from Fireworks
- Work with the new Easing In/Easing Out Controls

Unit Topics:

- Working with Blends
- Programming Blends
- Working with Filters
- Programming Filters
- Importing From Fireworks
- Using the New Easing In/Easing Out Controls

Working with Blends

Blend Modes

Blend two movie clips (or buttons) together using the following modes: layer, darken, multiply, lighten, screen, overlay, hardlight, add, subtract, difference, invert, alpha, and erase. Again, straight out of Photoshop. Graphic designers will love this feature, but not as much as filters :).

In Flash, blend modes let you create composite images. Compositing is the process of varying the transparency or color interaction of two or more overlapping objects. Blending modes also add a dimension of control to the opacity of objects and images. You can use Flash blending modes to create highlights or shadows that let details from an underlying image show through, or to colorize a desaturated image.

About blend modes

Blend modes let you create composite images. **Compositing** is the process of varying the transparency or color interaction of two or more overlapping objects. Blending lets you create unique effects by blending the colors in overlapping movie clips.

A blending mode contains these elements:

- **Blend color** is the color applied to the blend mode.
- **Opacity** is the degree of transparency applied to the blend mode.
- **Base color** is the color of pixels underneath the blend color.
- **Result color** is the result of the blend's effect on the base color.

Blend modes in Flash

Since blend modes depend on both the color of the object you're applying the blend to and the underlying color, you must experiment with different colors to see what the result will be. Macromedia recommends that you experiment with the different blend modes to achieve the effect you want.

Flash provides the following blend modes:

Normal applies color normally, with no interaction with the base colors.

Layer lets you stack movie clips on top of each other without affecting their color.

Darken replaces only the areas that are lighter than the blend color. Areas darker than the blend color don't change.

Multiply multiplies the base color by the blend color, resulting in darker colors.

Lighten replaces only pixels that are darker than the blend color. Areas lighter than the blend color don't change.

Screen multiplies the inverse of the blend color by the base color, resulting in a bleaching effect.

Overlay multiplies or screens the colors, depending on the base colors.

Hard light multiplies or screens the colors, depending on the blend mode color. The effect is similar to shining a spot light on the object.

Difference subtracts either the blend color from the base color or the base color from the blend color, depending on which has the greater brightness value. The effect is similar to a color negative.

Invert inverts the base color.

Alpha applies an alpha mask.

NOTE:

Alpha blend mode requires that a Layer blend mode be applied to the parent movie clip. You cannot change the background clip to Alpha and apply it because the object would appear invisible.

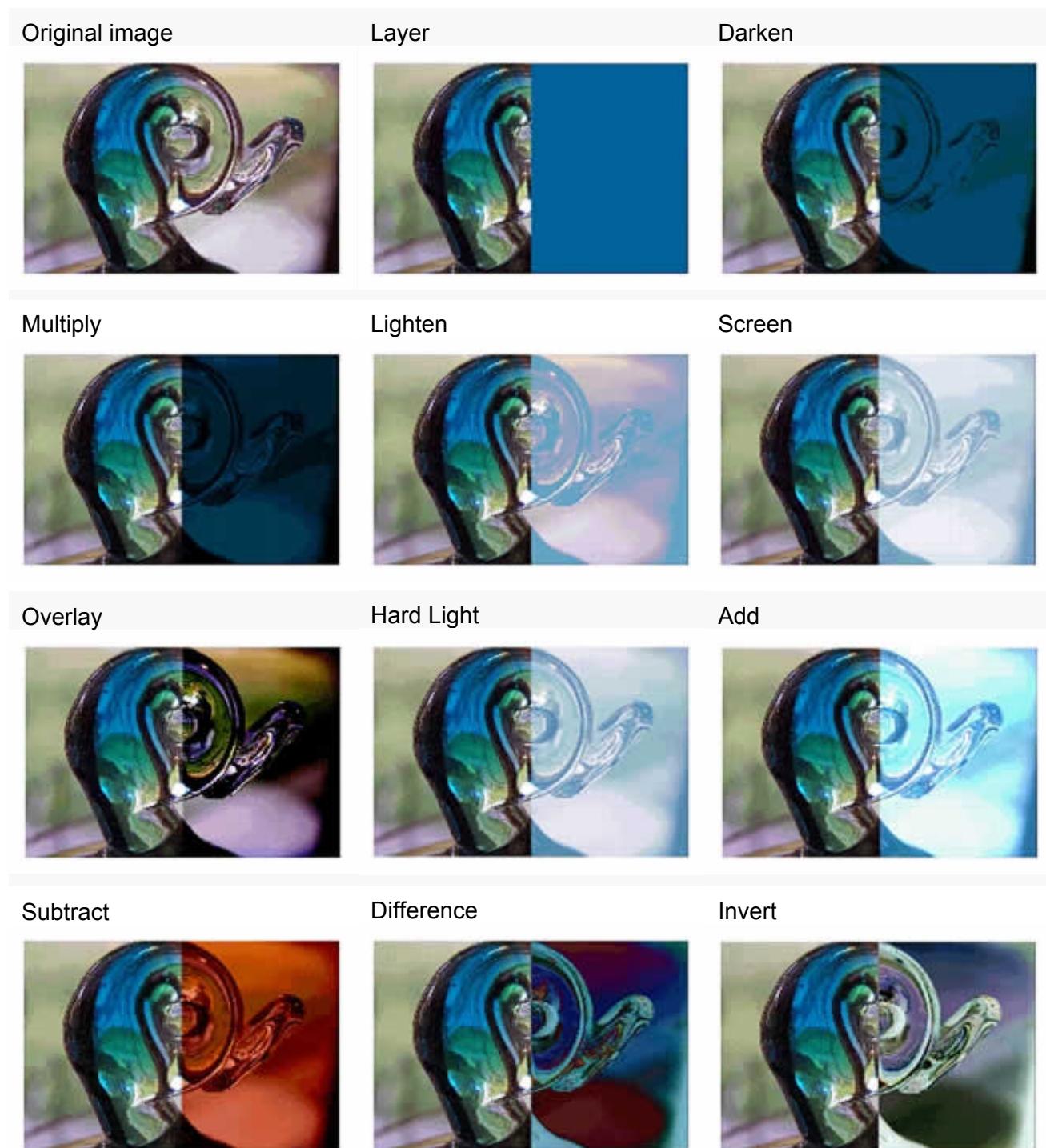
Erase removes all base color pixels, including those in the background image.

NOTE:

Erase blend mode requires that a Layer blend mode be applied to the parent movie clip. You cannot change the background clip to Erase and apply it because the object would appear invisible.

Blend mode examples

The following examples illustrate how different blend modes affect the appearance of an image. Be aware that the resulting effect of a blend mode may be considerably different, depending on the color of the underlying image and the type of blend mode you apply.



Applying a Blend Mode

You use the Property inspector for movie clips to apply blends to selected movie clips.

NOTE:

Multiple graphic symbols are merged as a single shape when you publish the SWF. For this reason, you cannot apply different blend modes to different graphic symbols.

To apply a blend mode to a movie clip:

1. Create a new Document and save it as **blend_test.fla**. Draw a red square and convert it to a movie clip. Draw a blue circle beside the red square and convert it to a movie clip.
2. Select the blue circle movie clip instance.
3. Adjust the color and transparency of the movie clip instance using the Color pop-up menu in the Property inspector.
4. Select a blend mode from the Blend pop-up menu in the Property inspector for movie clips. The blend mode is applied to the selected movie clip instance.



5. Position the movie clip with the blend mode over the graphic symbol whose appearance you want to modify.
6. Verify that the blend mode you selected is appropriate to the effect you're trying to achieve.

You might need to experiment with both the color and transparency settings of the movie clip and the different blend modes to achieve the effect you want.

Programming Blends

About blending modes

You can apply blend modes to movie clip objects by using the Flash workspace (Flash Professional 8) or ActionScript (Flash Basic 8 and Flash Professional 8). At runtime, multiple graphics are merged as one shape. For this reason, you cannot apply different blend modes to different graphic symbols.

Blend modes involve combining the colors of one image (the base image) with the colors of another image (the blend image) to produce a third image. Each pixel value in an image is processed with the corresponding pixel value of the other image to produce a pixel value for that same position in the result.

The `MovieClip.blendMode` property supports the following blend modes:

- **add** Commonly used to create an animated lightening dissolve effect between two images.
- **alpha** Commonly used to apply the transparency of the foreground on the background.
- **darken** Commonly used to superimpose type.
- **difference** Commonly used to create more vibrant colors.
- **erase** Commonly used to *cut out* (erase) part of the background using the foreground alpha.
- **hardlight** Commonly used to create shading effects.
- **invert** Used to invert the background.
- **layer** Used to force the creation of a temporary buffer for pre-composition for a particular movie clip.
- **lighten** Commonly used to superimpose type.
- **multiply** Commonly used to create shadows and depth effects.
- **normal** Used to specify that the pixel values of the blend image override those of the base image.
- **overlay** Commonly used to create shading effects.
- **screen** Commonly used to create highlights and lens flares.
- **subtract** Commonly used to create an animated darkening dissolve effect between two images.

Applying Blending Modes with ActionScript

The following procedure loads a dynamic image and lets you apply different blend modes to the image by selecting a blending mode from a combo box on the Stage.

To apply different blending modes to an image:

1. Create a new Flash document and save it as **blendmodes.fla**.
2. Drag a ComboBox component instance onto the Stage and give it an instance name of **blendMode_cb**.
3. Add the following ActionScript to Frame 1 of the Timeline:

```

var blendMode_dp:Array = new Array();
blendMode_dp.push({data:"add", label:"add"});
blendMode_dp.push({data:"alpha", label:"alpha"});
blendMode_dp.push({data:"darken", label:"darken"});
blendMode_dp.push({data:"difference", label:"difference"});
blendMode_dp.push({data:"erase", label:"erase"});
blendMode_dp.push({data:"hardlight", label:"hardlight"});
blendMode_dp.push({data:"invert", label:"invert"});
blendMode_dp.push({data:"layer", label:"layer"});
blendMode_dp.push({data:"lighten", label:"lighten"});
blendMode_dp.push({data:"multiply", label:"multiply"});
blendMode_dp.push({data:"normal", label:"normal"});
blendMode_dp.push({data:"overlay", label:"overlay"});
blendMode_dp.push({data:"screen", label:"screen"});
blendMode_dp.push({data:"subtract", label:"subtract"});
blendMode_cb.dataProvider = blendMode_dp;

var mcListener:Object = new Object();
mcListener.onLoadInit = function(target_mc:MovieClip) {
    var blendModeClip:MovieClip =
    target_mc.createEmptyMovieClip("blendModeType_mc", 20);
    with (blendModeClip) {
        beginFill(0x999999);
        moveTo(0, 0);
        lineTo(target_mc._width / 2, 0);
        lineTo(target_mc._width / 2, target_mc._height);
        lineTo(0, target_mc._height);
        lineTo(0, 0);
        endFill();
    }
    target_mc._x = (Stage.width - target_mc._width) / 2;
    target_mc._y = (Stage.height - target_mc._height) / 2;
    blendModeClip.blendMode = blendMode_cb.value;
};

this.createEmptyMovieClip("img_mc", 10);
var img_mcl:MovieClipLoader = new MovieClipLoader();
img_mcl.addListener(mcListener);
img_mcl.loadClip("http://www.helpexamples.com/flash/images/image1.jpg",
img_mc);

function cbListener(eventObj:Object):Void {
    img_mc.blendModeType_mc.blendMode = eventObj.target.value;
}
blendMode_cb.addEventListener("change", cbListener);

```

This ActionScript code populates the combo box with each type of blending mode, so the user can view each effect on the dynamically loaded image. A listener object is created, which is used with a MovieClipLoader instance. The listener object defines a single event listener, onLoadInit, which is invoked when the image is completely downloaded and is initialized by Flash.

The event listener creates a new movie clip named blendModeType_mc, and uses the Drawing API to draw a rectangular shape over the left half of the image. The currently selected blending mode for the ComboBox instance is then applied to the blendModeType_mc movie clip.

The rest of the code sets up the MovieClipLoader instance, which is responsible for loading the specified image into a movie clip on the Stage. Finally, a listener is defined for the blendMode_cb ComboBox instance, which applies the selected blending mode whenever a new item is selected from the ComboBox instance.

4. Select **Control > Test Movie** to test the document.

Working with Filters

About Filters

Filters let you add interesting visual effects to text, buttons, and movie clips and are most often associated with applying drop shadows, blurs, glows, and bevels to graphic elements. A feature unique to Flash is that you can animate the filters you apply using motion tweens. For example, if you create a ball (or sphere) with a drop shadow, you can simulate the look of the light source moving from one side of the object to another by changing the position of the drop shadow from its beginning and ending frames in the Timeline.

After you apply a filter, you can change its options any time, or rearrange the order of filters to experiment with a combined effect. You can enable or disable filters or delete them in the Property inspector. When you remove a filter, the object returns to its previous appearance. You can view the filters applied to an object by selecting it; doing so automatically updates the filters list in the Property inspector for the selected object.

About importing filters and blends from Fireworks PNG files (Flash Professional only)

Flash Professional 8 provides support for Fireworks filters and blending modes. When you import Fireworks PNG files, you can retain many of the filters and blending modes applied to objects within Fireworks, and continue to modify those filters and blends using Flash Professional 8.

Flash only supports modifiable filters and blends for objects imported as text and movie clips. If an effect or blend mode is not supported, Flash will rasterize or ignore it when it is imported. If you want to import a Fireworks PNG file that contains filters or blends that Flash does not support, you must rasterize the file during the import process. After this operation, you will not be able to edit the file.

Animating Filters

You can animate filters in the Timeline. Objects on separate keyframes joined by a tween have the parameters for corresponding filters tweened on intermediate frames. If a filter does not have a matching filter (a filter of the same type) at the opposite end of the tween, a matching filter is added automatically to ensure that the effect is at the end of the animation sequence.

Flash does the following to prevent motion tweens from functioning incorrectly in the event of a missing filter at one end of the tween, or filters applied in a different order at each end:

- If you apply a motion tween to a movie clip that has filters applied to it, when you insert a keyframe at the opposite end of the tween, the movie clip automatically has the same filters, with the same stacking order, on the last frame of the tween as it did at the beginning of the tween.
- If you place movie clips on two different frames with different filters applied to each, and you apply a motion tween between the frames, Flash first processes the movie clip with the most filters. Flash then compares the filters applied to the first movie clip against those used by the second movie clip.

If no matching filters are found in the second movie clip, Flash generates a "dummy" filter with no parameters and the color of the existing filters.

- If a motion tween exists between two keyframes:
 - If you add a filter to the object in one keyframe, Flash automatically adds a dummy filter to the movie clip when it reaches the keyframe at the other end of the tween.
 - If you remove a filter from an object in one keyframe, Flash automatically removes the matching filter from the movie clip when it reaches the keyframe at the other end of the tween.
- If you set filter parameters inconsistently between the beginning and end of a motion tween, Flash applies the filter settings of the starting frame to the interpolated frames. Inconsistent settings occur when the following parameters are set differently between the beginning and end of the tween: knockout, inner shadow, inner glow, and type of gradient glow and gradient bevel.

For example, if you create a motion tween using the drop shadow filter, and apply a drop shadow with a knockout on the first frame of the tween, and an inner shadow on the last frame of the tween, Flash corrects the inconsistent use of the filter in the motion tween. In this case, Flash applies the filter settings used on the first frame of the tween--a drop shadow with a knockout.

Filters and Flash Player performance

The type, number, and quality of filters you apply to objects can affect the performance of SWF files as you play them. The more filters you apply to an object, the greater the number of calculations Macromedia Flash Player must process to correctly display the visual effects you've created. For this reason, Macromedia recommends that you apply only a limited number of filters to a given object.

Each filter includes controls that let you adjust the strength and quality of the applied filter. Using lower settings improves performance on slower computers. If you are creating content for playback on a wide range of computers, or are unsure of the computing power available to your audience, set the quality level to low to maximize playback performance.

Applying Filters

You can apply one or more filters to selected objects using the Property inspector. Each time you add a new filter to an object, it is added to the list of applied filters for that object in the Property inspector. You can apply multiple filters to an object, as well as remove filters that have been previously applied.



The Add Filter menu in the Property inspector.

To apply a filter:

1. Select a movie clip, button, or text object on the Stage that you want to apply a filter to.
You can apply filters only to text, button, and movie clip objects.
2. Select the Filter tab in the Property inspector.
3. Click the Add Filter (+) button and select a filter from the Filters pop-up menu.

The filter you select is applied to the object, and the controls for the filter settings appear in the Property inspector. Experiment with the settings until you get the look you want. The different types of Filters include the following:

- Drop shadow
- Blur
- Glow
- Bevel
- Gradient Glow

- Gradient Bevel
- Adjust Color

To remove a filter:

1. Select the movie clip, button, or text object that you want to remove a filter from.
2. Select the Filter tab in the Property inspector.
3. Select the filter you want to remove in the list of applied filters.
4. Click the Remove Filter (-) button to remove the filter.

You can create a filter settings library that lets you easily apply the same filter or sets of filters to an object. Flash stores the filter presets you create in the Property inspector on the Filters tab in the Filters > Presets menu.

You can delete or rename any presets.

To apply a preset filter to an object:

1. Select the movie clip, button, or text object that you want to apply a filter preset to.
2. Select the Filter tab in the Property inspector.
3. Click the Add Filter (+) button, and select Presets from the Filters pop-up menu.
4. Select the filter preset you want to apply from the list of available presets at the bottom of the preset menu.

NOTE:

When you apply a filter preset to an object, Flash replaces any filters currently applied to the selected object(s) with the filter(s) used in the preset.

To enable or disable a filter applied to an object:

- Click the enable or disable icon next to the filter name in the Filter list in the Property inspector.

NOTE:

Alt-click (Windows) or Option-click (Macintosh) the enable icon in the Filter list to toggle the enable state of the other filters in the list. If you Alt-click the disable icon, the selected filter is enabled, and all others filters in the list are disabled.

To enable or disable all filters applied to an object:

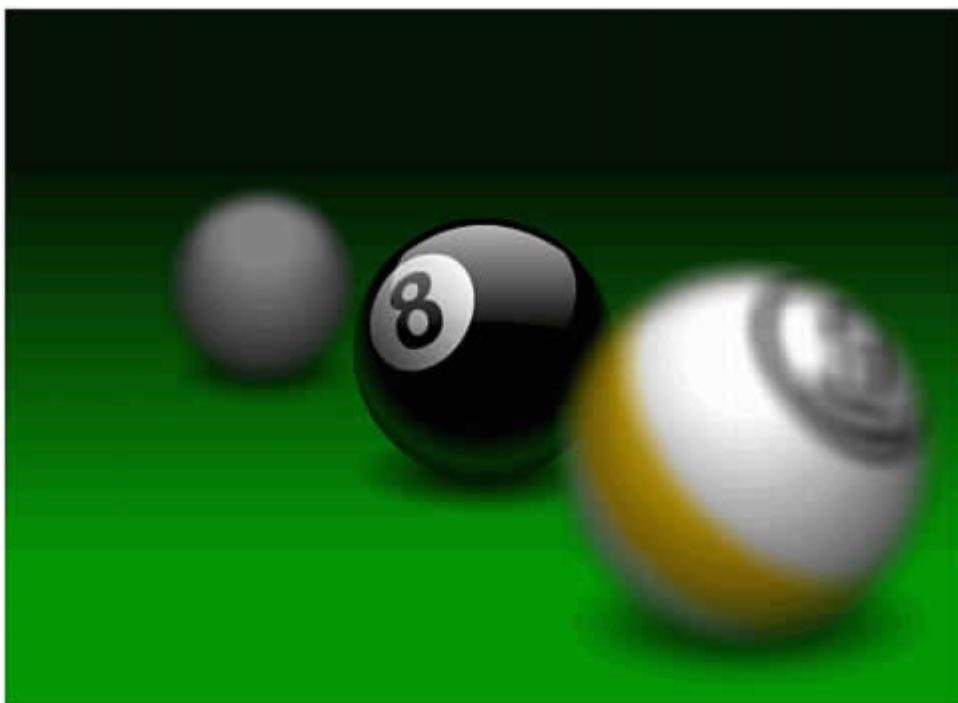
- Click the Add Filter (+) button in the Property inspector, and then select Enable All or Disable All from the pop-up menu.

Apply Graphic Filters and Blends

This tutorial guides you through the process of creating eye-catching graphic effects using some of the authoring features in Macromedia Flash Professional 8 (Filters are not available in Flash Basic). By using the graphic filters and blend modes available in Flash, you can transform ordinary graphic objects into much more visually compelling content.

Review your task

In this tutorial, you will add some graphics to the Stage and apply filters to them to create a realistic-looking image of a pool table with a narrow depth of field. The balls in the foreground and background will appear out of focus, as if seen through a camera.



The completed FLA file

To see the completed FLA file (**Filters&Blends_finished.fla**), browse to one of the following locations:

- In Windows: *Unit_5/*Filters and Blends
- On the Macintosh: *Unit_5/*Filters and Blends

Open the starter document

Now that you have seen the document you will create, it is time to create your own version of the document. The first thing to do is open the starter document, which contains the graphic objects you will use with the filter and blend features of Flash.

1. In Flash, select File > Open.
2. Browse to one of the following locations:
 - In Windows: *Unit_5\Filters and Blends*
 - On the Macintosh: *Unit_5/Filters and Blends*
3. Select the file named **Filters&Blends_start.fla** and click Open.

Apply filters and blends

In the following sections, you will apply blends and filters to the 9ball and CueBall movie clips found in the Library panel.

Apply a blend to the 9ball movie clip

The first task you must perform in the starter file is to apply a blend effect to an instance of the 9ball outlines movie clip to make its colors combine with the colors of the 9ball movie clip. This achieves a realistic shiny pool ball look.

1. Open the **Library panel** (**Window > Library**).
2. Drag the **9ball** movie clip from the Library panel to the lower-right corner of the Stage.

This creates an new instance of the 9ball movie clip on the Stage.
3. **Double-click** the new 9ball instance to enter symbol-editing mode.
4. Drag the 9ball outlines movie clip from the Library panel onto the Stage so that it perfectly covers the 9ball instance.

You can use the Arrow keys to make fine adjustments to the placement of the 9ball outlines instance.

5. With the 9ball outlines instance still selected, go to the Property inspector and select **Multiply** from the Blend menu.
6. Double-click outside the Stage to exit symbol-editing mode.

The 9ball instance should still be selected on the Stage.
7. Open the **Transform panel** (**Windows > Transform**).
8. In the Transform panel, click the **Constrain** check box.
9. Double-click the Width text box and enter **140**.

10. This makes the 9ball instance appear larger than the 8ball instance on the Stage.
11. Reposition the 9ball instance so it slightly overlaps the right edge of the 8ball instance on the Stage.

Apply a filter

The next step is to apply a blur filter to the 9ball movie clip to make it appear out of focus, as if it is in the foreground of the Stage.

1. With the 9ball instance still selected, click the **Filters tab** in the Property inspector.
2. In the **Filters tab**, click **Add Filter** and select **Blur** from the pop-up menu.
3. Drag the BlurX slider until the BlurX and BlurY value is **19**.

The BlurX and BlurY values are constrained to each other by default. Click the lock icon if they are not.

4. Click the Properties tab of the Property inspector.
5. In the Property inspector, enter the following values in the W, H, X, and Y text boxes:

W: 210

H: 235

X: 315

Y: 155

6. Click outside the Stage to deselect the 9ball instance.

Apply filters and blends to the cue ball

The last step is to use a color effect and a filter to make the cue ball appear to be in the background of the Stage.

1. Drag the CueBall movie clip from the Library panel to the upper-left corner of the Stage.
2. With the new CueBall instance still selected on the Stage, go to the Transform panel and enter **70** in the W (Width) and H (Height) text boxes.

This makes the CueBall instance appear smaller than the 8ball instance.

3. With the CueBall instance still selected on the Stage, go to the Property inspector and select Brightness from the Color menu.
A Brightness Amount slider appears to the right of the Color menu.
4. Drag the Brightness Amount slider until the value is **-48%**.

5. Enter the following values in the W, H, X, and Y text boxes:

W: **105**

H: **115**

X: **95**

Y: **105**

6. With the CueBall instance still selected on the Stage, click the Filters tab in the Property inspector.
7. Click **Add Filter** and select **Blur** from the Filter pop-up menu.
8. Drag the BlurX slider until the BlurX and BlurY values are each **13**.

These text boxes are constrained to match each other by default. Click the lock icon if they are not.

9. Click outside the Stage to deselect the CueBall instance.

You have now applied the graphic effects to create a realistic look of depth.

Programming Filters

Working with filters using ActionScript

The `flash.filters` package contains classes for the bitmap filter effects that are new in Flash Player 8. Filters let you use ActionScript to apply rich visual effects, such as blur, bevel, glow, and drop shadow, to text, movie clip, and button instances. You can also use the Flash authoring tool to apply filter effects to objects such as text, images, and video. Flash has nine filter effects, although only seven are accessible by using the user interface in Flash Professional 8. The `ConvolutionFilter` and `DisplacementMapFilter` filters are only available by using ActionScript code.

NOTE:

All filters are available by using ActionScript in both Flash Basic 8 and Flash Professional 8

The following procedure loads a semitransparent PNG image, and applies a `GlowFilter` effect to the nontransparent portion of the image.

To apply filters to semitransparent images:

1. Create a new Flash document and save it as **transparentImg.fla**.
2. Add the following ActionScript to Frame 1 of the Timeline:

```
import flash.filters.GlowFilter;
System.security.allowDomain("http://www.helpexamples.com");
var mclListener:Object = new Object();
mclListener.onLoadInit = function(target_mc:MovieClip) {
    target_mc._x = (Stage.width - target_mc._width) / 2;
    target_mc._y = (Stage.height - target_mc._height) / 2;
    var glow:GlowFilter = new GlowFilter();
    target_mc.filters = [glow];
};
this.createEmptyMovieClip("img_mc", 10);
var img_mcl:MovieClipLoader = new MovieClipLoader();
img_mcl.addListener(mclListener);
img_mcl.loadClip("http://www.helpexamples.com/flash/images/logo.png", img_mc);
```

This code uses a movie clip loader instance to load a semi-transparent PNG image. After the image finishes loading, the image moves to the center of the Stage and a glow filter is applied.

3. Select **Control > Test Movie** to test the document.

The glow filter effect is only applied to the opaque (nontransparent) area of the PNG image.

Importing From Fireworks

Fireworks effects supported in Flash

The following Fireworks effects are imported as modifiable filters by Flash:

Fireworks Effect	Flash Filter
Drop shadow	Drop shadow
Solid shadow	Drop shadow
Inner shadow	Drop shadow (with Inner shadow automatically selected)
Blur	Blur (where blurX = blurY=1)
Blur more	Blur (where blurX = blurY=1)
Gaussian blur	Blur
Adjust color brightness	Adjust color
Adjust color contrast	Adjust color

Blend modes

You can achieve a variety of compositing effects by using blend modes to change the way the image of one object on the Stage is combined with the images of any objects beneath it.

Fireworks blend modes supported in Flash

Flash imports the following Fireworks blend modes as modifiable blends:

Fireworks Blending Mode	Flash Blending Mode
Normal	Normal
Darken	Darken
Multiply	Multiply
Lighten	Lighten
Screen	Screen
Overlay	Overlay
Hard light	Hard light
Additive	Add
Difference	Difference
Invert	Invert
Alpha	Alpha
Erase	Erase

Flash ignores all other blending modes imported from Fireworks. The blending modes that are not supported in Flash are Average, Negation, Exclusion, Soft Light, Subtractive, Fuzzy Light, Color Dodge, and Color Burn.

Using the New Easing In/Easing Out Controls

A *tween* is the application of a change to a graphic object over a period of time. For example, you could tween the location of a picture of a car from one side of the Stage to the other to make the car move from side to side. To *ease* a tween is to control the rate at which the change is applied to the object. New easing controls in Flash allow you to precisely control how tweens that you apply in the timeline affect the appearance of tweened objects on the Stage. With the new controls, you can make an object move back and forth on the Stage within a single tween or create other complex tween effects.

Custom Easing Controls with Various Properties

You have complete control over acceleration and deceleration for motion tweens. You can also set easing for position, rotation, scale, color, and even filters separately. This is really an amazing feature, and it will speed along animation development.

Applying custom ease in/ease out to motion tweens

The Custom Ease In/Ease Out dialog box displays a graph representing the degree of motion over time. Frames are represented by the horizontal axis, and the percentage of change is represented by the vertical axis. The first keyframe is represented as 0%, and the last keyframe is represented as 100%.

The rate of change of the object is represented by the slope of the graph's curve. When the curve is horizontal (no slope), the velocity is zero; when the curve is vertical, there is an instantaneous rate of change.

The dialog box provides the following additional controls:



Use One Setting for All Properties check box

The default value for this is selected, which means that the displayed curve is used for all properties, and the Properties pop-up menu is disabled. When the check box is not selected, the Properties pop-up menu is enabled, and each property has a separate curve defining the velocity of that property.

Property pop-up menu

This menu is enabled only when the Use One Setting for All Properties check box is not selected. When enabled, a separate curve is maintained for each of the five properties appearing in the menu. Selecting a property in the menu displays the curve for that property. The properties are:

Position

Specifies custom ease settings for the position of an animated object on the Stage.

Rotation

Specifies custom ease settings for the rotation of an animated object. For example, you can fine-tune how quickly or slowly an animated character turns around to face the user on the Stage.

Scale

Specifies custom ease settings for the scale of an animated object. For example, you can more easily customize the scale of an object so it appears to be moving away from the viewer, then coming closer, and then moving away again.

Color

Specifies custom ease settings for color transitions applied to an animated object.

Filters

Specifies custom ease settings for filters applied to an animated object. For example, you can control the ease setting of a drop shadow that simulates a change in the direction of a light source.

Play and Stop buttons These buttons let you preview an animation on the Stage using all the current velocity curves defined in the Custom Ease In/Ease Out dialog box.

Reset button

This button lets you reset the velocity curve to the default, linear state.

Position of the selected control point In the lower-right corner of the dialog box, a numeric value displays the keyframe and position of the selected control point. If no control point is selected, no value appears.

To add a control point to the line, click the diagonal line once. This adds a new control point to the line. By dragging the positions of the control points, you can achieve a precise degree of control over the motion of an object.

Using frame indicators (represented by square handles), you click where you want an object to slow down or speed up. Clicking the handle of a control point (the square handles) selects that control point, and displays the tangent points on either side of it. Tangent points are represented by hollow circles. You can drag the control point or its tangent points with the mouse or you can position them using the keyboard's arrow keys.

TIP:

By default, the control points snap to a grid. You can turn off snapping by pressing the X key while dragging the control point.

Clicking an area of the curve away from any control points adds a new control point to the curve at that point, without changing the shape of the curve. Clicking away from the curve and control points deselects the control point that is currently selected.

Compatibility with current ease in/ease out settings

If you apply a custom ease to a frame using the Custom Ease In/Ease Out dialog box, the edit text box that displays the ease value displays '--'. If you apply an ease value to a frame with the Edit text box or the pop-up slider, the Custom Ease graph is set to the equivalent curve, and the Use One Setting for All Properties check box is selected.

Unsupported easing curves

Certain types of easing curves are not supported:

- No part of the graph can represent a nonlinear curve (such as a circle)
- The Custom Ease dialog box automatically prevents moving a control point or a tangent handle to a position that would render an invalid curve.
- All points must exist on the graph. Control points cannot be moved beyond the bounds of the graph.
- All segments of the curve must exist within the graph. The shape of the curve will be flattened to prevent it from extending beyond the bounds of the graph.

To use the Custom Ease In/Ease Out dialog box:

1. Select a layer in the Timeline that has a motion tween applied to it.
2. Click the Edit button next to the Ease slider in the frame Property inspector. The Custom Ease In/Ease Out dialog box appears.
3. (Optional) Deselect the Use One Setting for All Properties check box, and select a property in the menu to display the curve for that property. For more information about the properties you can specify, see Applying custom ease in/ease out to motion tweens (Flash Professional only).
4. In the Custom Ease In/Ease Out dialog box, Control-click (Windows) or Command-click (Macintosh) on the diagonal line to add a control point.
5. Drag the control point up to increase the speed of the object, or drag it downwards to slow down the speed of the object.
6. Drag the vertex handles to further adjust the ease curve, and fine tune the ease value of the tween.
7. View the animation on the Stage by clicking the play button in the lower left corner of the Custom Ease In/Ease Out dialog box.
8. Adjust the controls until you achieve the effect you want.

To copy and paste an ease curve:

- Copy the current ease curve by pressing Control+C (Windows) or Command+C (Macintosh)
- Paste the copied curve into another ease curve by pressing Control+V (Windows) or Command+V (Macintosh)

You can copy and paste the ease curve. The copied curve remains available until you exit the Flash application.

Follow the Bouncing Text

Well, let's get rolling on our animated bouncing text effect. To get started, just open Flash 8 and create a new document. I've also provided the finished FLA file for your reference; you can download it from the sample files download link at the beginning of this article.

Creating the Text

Launch Flash 8 and create a new Flash document from the start screen. I named my file `bouncing_text.fla`, but you can name your file whatever you like.

- Create a new Symbol in Flash. (Select Insert > New Symbol.) Name the symbol `gr_text` and select Graphic as the behavior type in the Create New Symbol dialog box. Click the OK button.

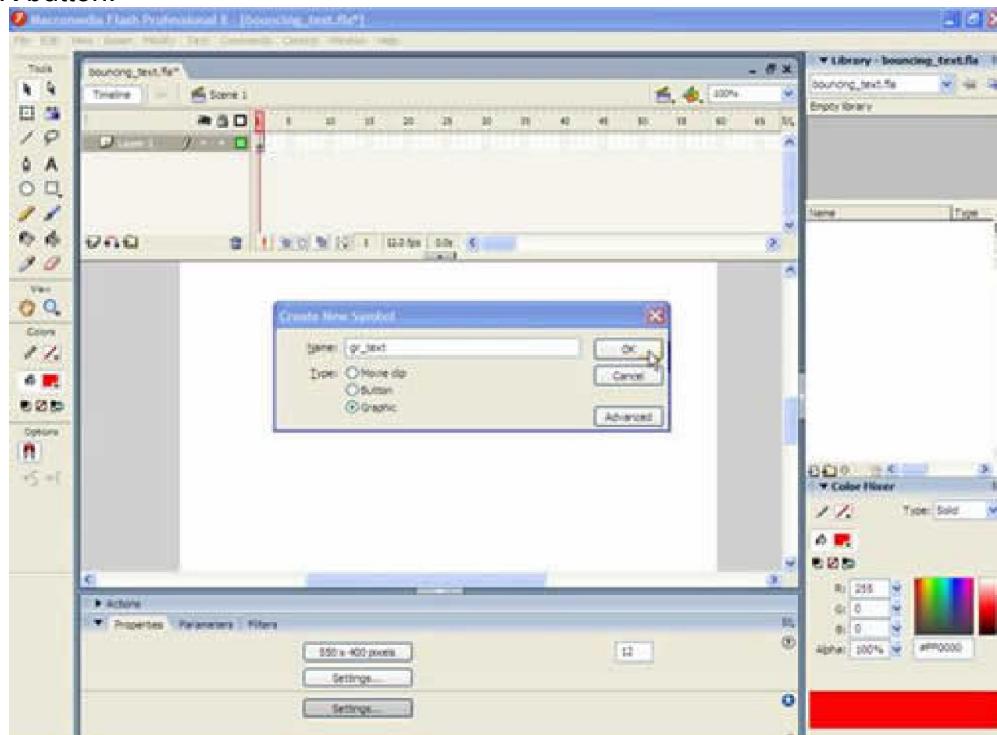


Figure 1. Creating the new text symbol

Now you'll create the text for the animation. In the Tools panel, select the Text tool (T).

1. In the Properties panel, make sure that Static Text is selected. Select a bold font like **Arial Black**, but you can pick any font you like. Next, set the font size to **80** and set the text fill color to **#FF0000** (Red).
2. Still in the Properties panel, change the Anti-alias option to **Anti-Alias for animation**. This setting improves the overall motion quality of animated text, and its visibility.
3. Draw a text area in the middle of the Stage by clicking and dragging with the Text tool (T).
4. Type in the text **Flash 8**. (See Figure 2.) Notice that in Flash 8 the text area now has control points on all four corners. You can adjust the width of your text box by selecting any one of the edit nodes and dragging it to the left or right.

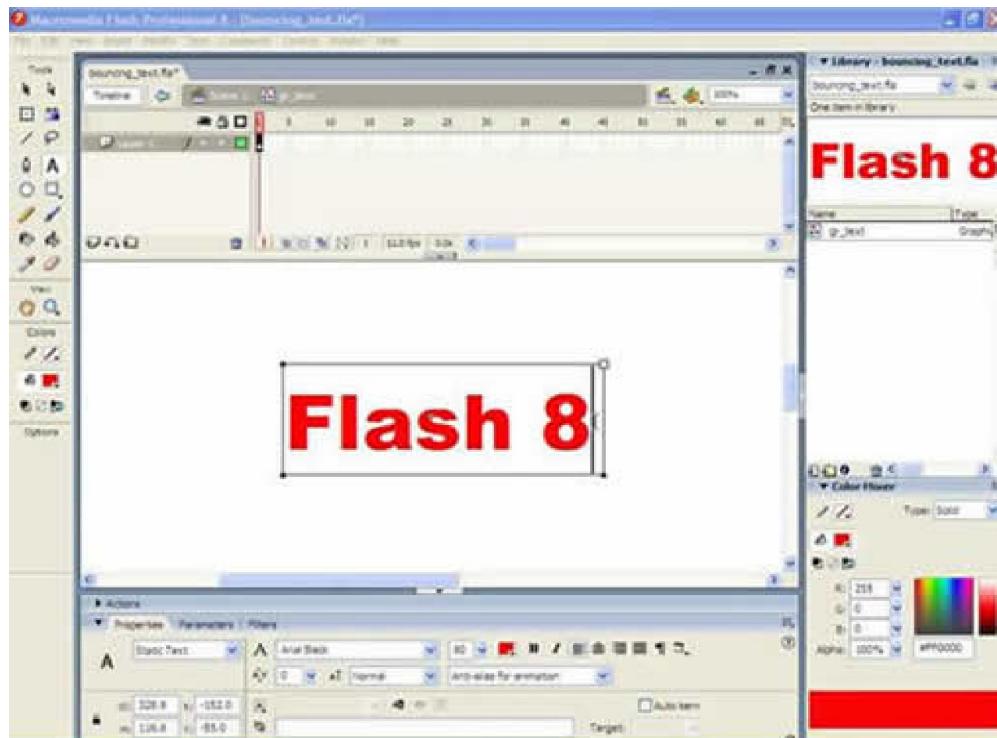


Figure 2. The new Static Text area with the Anti-alias for animation setting

Adding the Bevel Filter

In these next steps we will be adding a Bevel Filter and modifying the Filter's attributes.

Select the **Filters** tab in the Property Inspector. Filters are another great new feature in Flash 8 that will allow you to add shadows, glows, and other visual effects to your objects. For our text, we're going to use a bevel filter.

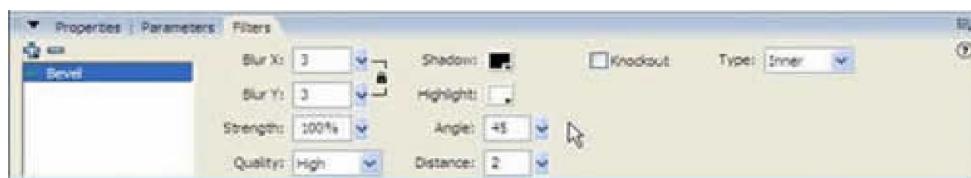


Figure 3. Setting the Bevel Filter attributes in the Property inspector

1. Click the **Add** icon to expose the filter menu. Select the **Bevel** option. You can add additional effects to your objects. For this project, we'll stick with just one.
2. Change the following Bevel attributes (See Figure 3.):
 - **Blur X: 3**
 - **Blur Y: 3**
 - **Quality: High**
 - **Distance: 2**
3. Keep the other settings to their default values.
4. Notice that the text now takes on a 3-D effect - that was too easy. Having the ability to add these Filters in Flash 8 will save you plenty of time. (See Figure 4.)
5. Finally, select Scene 1 in Timeline panel to jump out of the gr_text symbol. Now we are ready to animate our text with a motion tween.

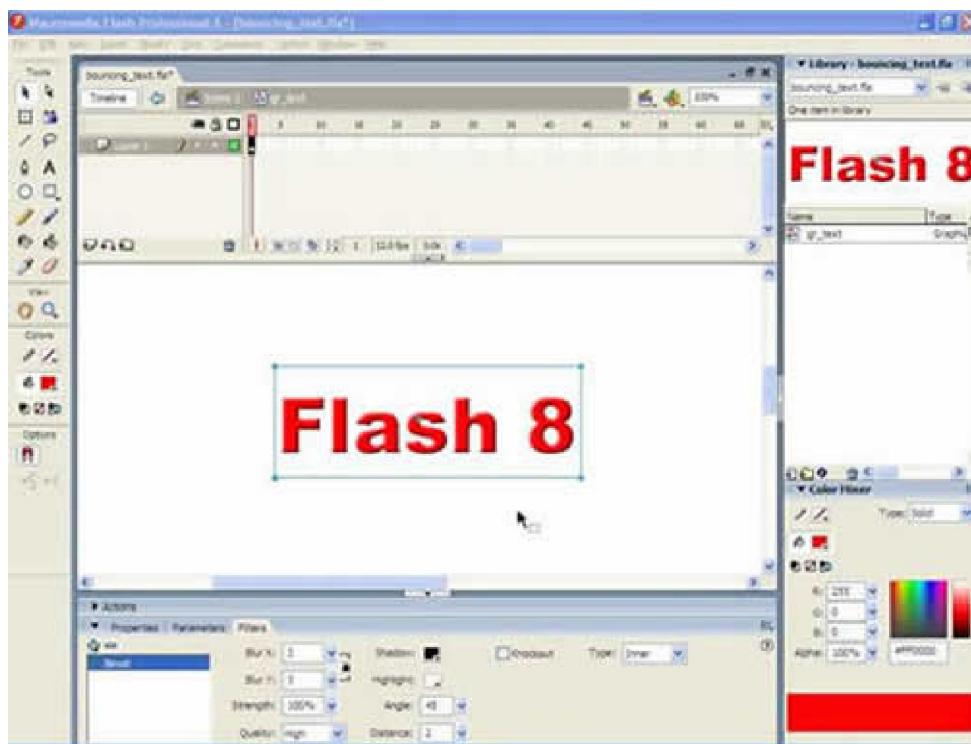


Figure 4. The text now has a Bevel Filter applied to it

Create the Motion Tween

1. In the tools panel, choose the **Selection tool** (V). In the Timeline double click the name "Layer 1" to rename it to Flash Text.
2. From the Library Panel, drag and instance of the **gr_text** symbol to the stage. At this point we have our symbol on frame 1.
3. Next let's add a keyframe to frame 30. Select frame 30 on the Flash Text layer. From the Insert menu select (**Insert > Timeline > Keyframe**).

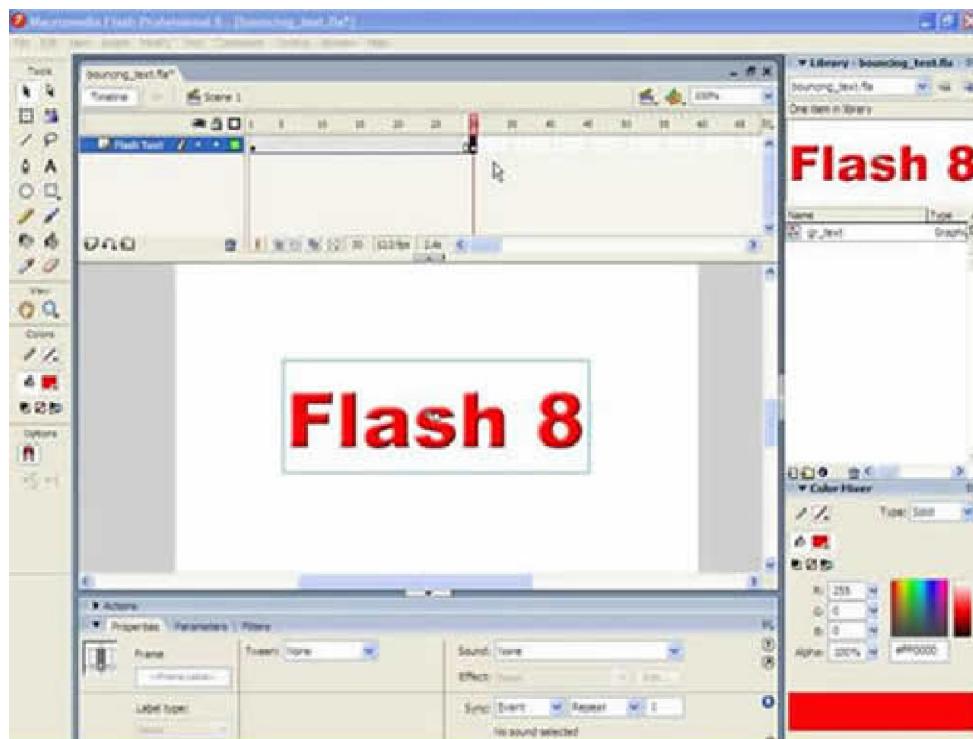


Figure 5. Place the symbol on the stage and add a keyframe

4. Select **Frame 1** in the Flash Text layer.
5. In the Property Inspector, choose the Motion option from the Tween drop down menu. We don't have an animation just yet. We need to make another adjustment to Frame 1.
6. While still in Frame 1 take your **Selection tool** (V) and click on the graphic symbol to select it.
7. We are going to scale the text down by selecting (**Modify > Transform > Scale and Rotate...**)
8. In the Scale and Rotate dialog box, set the scale value to **10%** and click the **OK** button. At this point you can now click on the playback head in the timeline and scrub back and forth to see the animation effect.

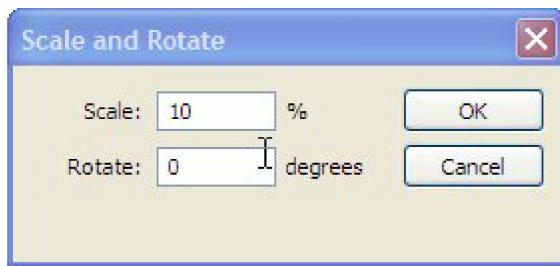


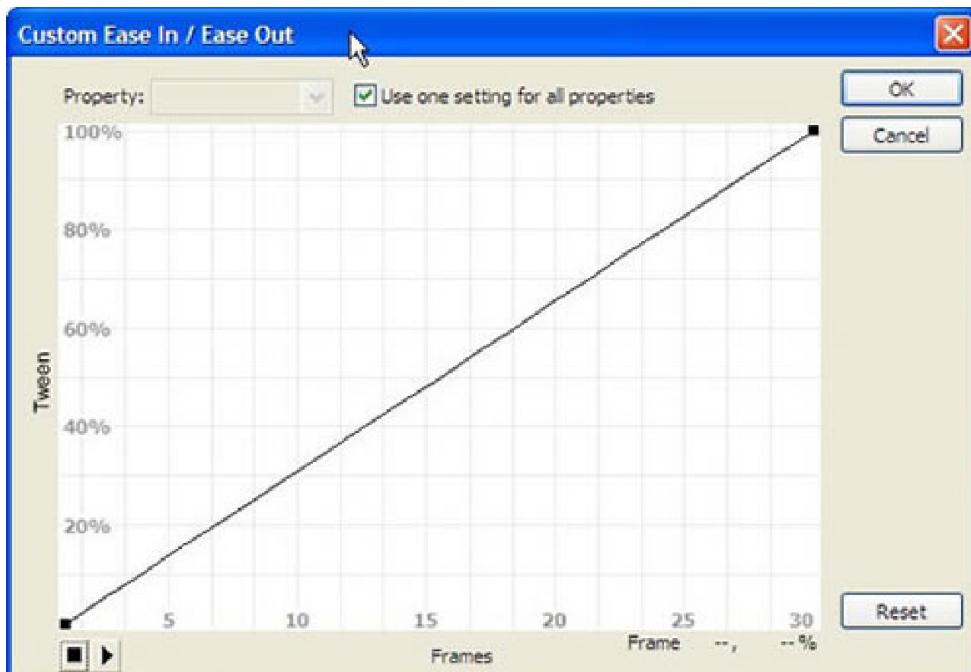
Figure 6. Modify the scale of the symbol in frame one

9. Test your movie by selecting the Control menu (**Control > Test Movie**). You will notice that in Flash 8 that now when you test your .swf file – a Flash Player window will appear. This will give you a better representation of your final output.
10. Close the test movie by clicking the close window button. Well, I have to admit that this animation is really nothing to write home about. That is all going to change when we add a “bounce” effect to our animation.

Using Custom Ease In / Ease Out

Finally, we are going to modify our existing animation by giving it a “bounce” effect with the new Ease In / Ease Out controls. This is another great new feature in Flash 8 that will add sophistication to your animation without the use of additional keyframes.

11. With your Selection tool (V) choose **Frame 1** from the Flash Text layer.
12. In the Property Inspector click on the **Edit** button next to the Ease drop down menu.
13. The Custom Ease In / Ease Out dialog box appears. (See Figure 7.)



14. Figure 7. The Custom Ease In / Ease Out dialog box
15. Click the Ease In/Ease Out line to add and edit node at **Frame 10**.
16. Take the new node and drag it up to a Teen value of **80%**. This represents the completion value of the entire motion tween from frame 1 to frame 30.
17. Click the Ease In/Ease Out line to add a second edit node at **Frame 20**.
18. Take the new node and drag it down to a Teen value of **20%**. (See Figure 8.)

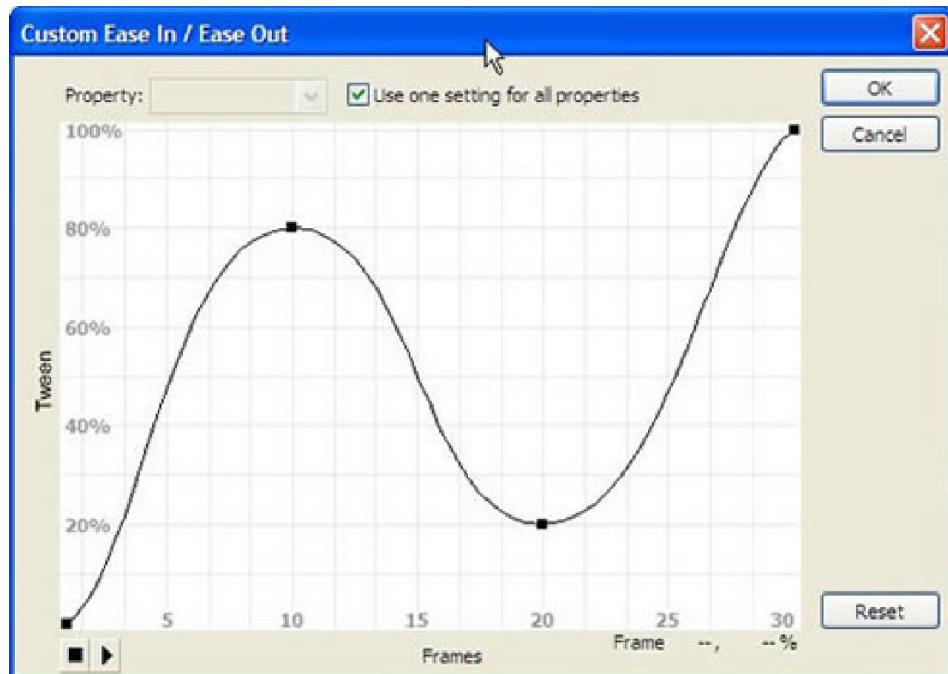


Figure 8. Modify the motion tween by adding edit nodes

19. Click the **Play** button in the bottom left corner of the dialog box to view your animation. Now you can see the “bounce” effect on your motion tween with no additional reframes added. That’s really cool.
20. Close the Ease In/Ease out dialog box and test your movie. You have now finished your bouncing text!

Lab

- Create your own Motion Tween and try the custom Easing In and Out features.

Summary of New Features

- New Blends modes
- You can program blends
- New Graphic Filters
- You can program Filters
- Importing From Fireworks retains Filter and Blend attributes
- New Easing In/Easing Out Controls

Unit 6: Text and Graphics Rendering

Unit Objectives:

After this unit, you will be able to:

- Use the New Text Handles
- Use the Various Text Anti-Alias Settings
- Use Bitmap Smoothing
- Understand How Bitmap Caching Works
- Load new Graphic Formats Dynamically
- Create a MovieClip with 9-Slice Scaling
- Understand the ConvolutionFilter
- Use Programmatic Bitmap Creation with ActionScript

Unit Topics:

- FlashType Overview
- FlashType Anti-Alias Settings
- Using Bitmaps in Flash 8
- Loading Graphics Dynamically
- Using the New 9-Slice Scaling Feature
- Using ActionScript to Modify Bitmaps

FlashType Overview

FlashType is a new text rendering engine that provides clear, high-quality text rendering both in the Flash authoring environment and in the published SWF files. FlashType greatly improves the readability of text, particularly when it is rendered at smaller font sizes. While FlashType is available in both Flash Basic and Flash Professional, the new custom anti-aliasing option is available only in Flash Professional. Custom anti-aliasing lets you specify the thickness and sharpness of fonts used in individual text fields.

FlashType is automatically enabled whenever Flash Player 8 is the selected version of the player, and Anti-Alias for Readability or Custom Anti-Alias is the selected anti-aliasing mode. Using FlashType may cause a slight delay when loading Flash SWF files. You will notice this delay especially if you are using several different character sets (four or five) within the first frame of a Flash document, so be aware of the number of fonts you use. FlashType font rendering may also cause an increase in the Flash Player's memory usage. Using four or five fonts, for example, can increase memory usage by approximately 4 MB.

When Flash Player 8 is the selected version of Flash Player, and Anti-Alias for Readability or Custom Anti-Alias is your chosen anti-aliasing option, FlashType anti-aliasing applies to the following:

- Untransformed text that is scaled and rotated

NOTE: While the text can be scaled or rotated, it must remain flat (for example, untransformed). For example, if you skew the fonts or otherwise manipulate the font shapes, FlashType automatically is disabled.

- All font families (including bold, italic, and so on)
- Display sizes up to a size of 255 points

NOTE: Because magnification affects the display size of the text, when you zoom in, the text redraws at larger point sizes, thereby disabling the anti-aliasing for readability font rendering once a size of 255 points is reached.

- Exporting to most non-Flash file formats (GIF or JPEG)

FlashType is disabled under the following conditions:

- Flash Player 7 or earlier is the selected version of Flash Player.
- An anti-aliasing option other than Anti-Alias for Readability or Custom Anti-Alias is selected.
- Text is skewed or flipped.
- The FLA file is exported to a PNG file.

FlashType Anti-Alias Settings

Anti-aliasing lets you smooth text so that the edges of characters displayed onscreen look less jagged. The anti-aliasing options make text more readable by aligning text outlines along pixel boundaries and is particularly effective for more clearly rendering smaller font sizes. When anti-aliasing is enabled, all text in the current selection is affected. Anti-aliasing operates with text of all point sizes in the same way.

Anti-aliasing is supported for static, dynamic, and input text if the user has Flash Player 7 or later. It is supported only for static text if the user has an earlier version of Flash Player.

A new feature in Flash 8 is FlashType, a new text rendering engine that provides improved anti-aliasing and readability of fonts.

When using small text in a Flash document, keep in mind the following guidelines:

- Sans serif text, such as Helvetica or Arial, appears clearer at small sizes than serif text.
- Some type styles, such as bold and italic, can reduce text readability at small sizes.
- In some cases, text appears somewhat smaller than text of the same point size in other applications.

Setting anti-aliasing options for text

Flash provides improved font rasterization that lets you specify the anti-aliasing properties for fonts. The improved anti-aliasing capabilities are available only when publishing SWF files for Flash Player 8. If you are publishing files for use with Flash Player 7 or earlier versions, you can only use the Anti-Alias for Animation feature.

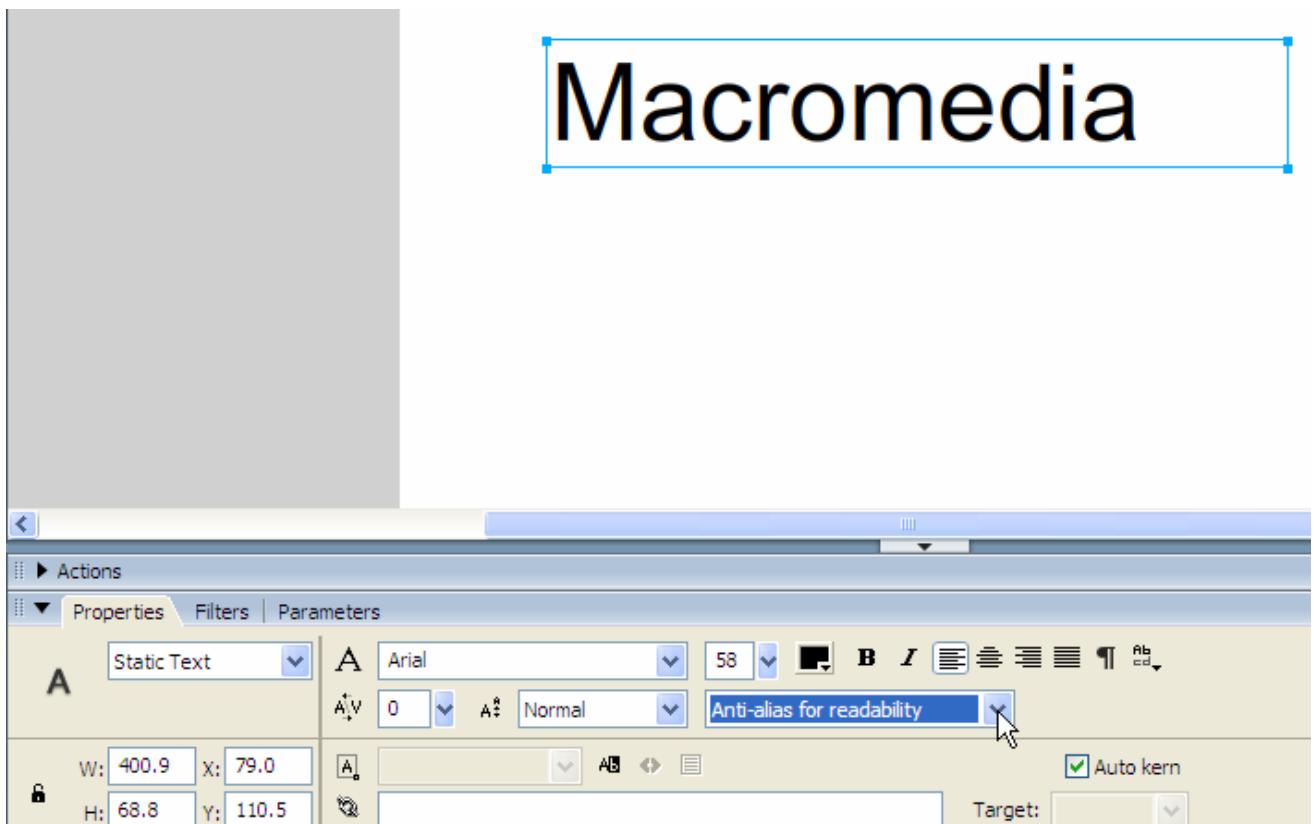
You apply anti-aliasing for each text field rather than each character. Also, when you open existing FLA files in Flash 8, the text is not automatically updated to the Advanced Anti-Alias option; you must select individual text fields and manually change the anti-aliasing settings. For more information, see Upgrading Flash 7 content to use Flash 8 anti-aliasing options.

To select an anti-aliasing option for selected text:

- In the Property inspector, choose one of the following options from the Anti-Aliasing pop-up menu:

Use Device Fonts specifies that the SWF file use the fonts installed on the local computer to display the fonts. Although this option has the least impact on the size of the SWF file, it also forces you to rely on the fonts installed on the user's computer for font display. For example, if you specify the font Times Roman as a device font, the Times Roman font must be installed on the computer playing back the content for the text to display. For this reason, when using device fonts, choose only commonly installed font families.

Bitmap Text (No Anti-Alias) turns off anti-aliasing and provides no text smoothing. The text is displayed using sharp edges, and the resulting SWF file size is increased because the font outlines are embedded in the SWF. Bitmap text is sharp at the exported size, but scales poorly.



Anti-Alias for Animation creates a smoother animation. This is possible in part because Flash ignores alignment and kerning information. Specifying Anti-Alias for Animation creates a larger SWF file, because font outlines are embedded.

NOTE: Fonts rendered using Anti-Alias for Animation are less legible at smaller font sizes. for this reason, it is recommended that you use 10 point or larger type when specifying Anti-Alias for Animation.

Anti-Alias for Readability uses a new anti-aliasing engine that improves the readability of fonts, particularly at small sizes. Specifying Anti-Alias for Readability creates a larger SWF file, because font outlines are embedded. In order to use the Anti-Alias for Readability setting you must publish to Flash Player 8.

NOTE: Anti-Alias for Readability creates a highly legible typeface even at small sizes. However, it animates poorly, and may cause performance problems. If you intend to animate text, use Anti-Alias for Animation.

Custom Anti-Alias (Flash Professional only) lets you modify the font's properties as you see fit. The custom anti-aliasing properties are as follows:

Sharpness determines the smoothness of the transition between the text edges and the background.

Thickness determines how thick the font anti-aliasing transition appears. Larger values cause the characters to look thicker.

Specifying Custom Anti-Alias creates a larger SWF file, because font outlines are embedded. In order to use the Custom Anti-Alias setting, you must publish to Flash Player 8 and author your Flash content using Flash Professional 8.

Upgrading Flash 7 content to use Flash 8 anti-aliasing options

When you open a FLA file created for use with Flash Player 7 or earlier, the text Property inspector sets the anti-alias option to its equivalent anti-aliasing option from Flash MX 2004. Text from older FLA files can be anything except for Anti-Alias for Readability and Custom Anti-Alias.

To upgrade Flash 7 content to use the Advanced or Custom Anti-Alias options:

Open a FLA file created for use with Flash 7 or earlier.

Open the Publish Settings dialog box (File > Publish Settings), and select Flash Player 8 from the Version pop-up menu.

Select the text field to which you want to apply the Anti-Alias for Readability or Custom Anti-Alias option.

In the Property inspector, select Anti-Alias for Readability or Custom Anti-Alias from the Font Rendering Method pop-up menu.

Adding text

You can add several types of text to a Flash document: static text, dynamic text, or input text. Static text is useful when you need to add decorative text to the Stage, or any text that doesn't need to change or load from an external source. Use dynamic text when you need to load text from a file, database, or change the text when the SWF file plays in Flash Player. Use input text when you want the user to type into a text field. You can take that text and send it to a database, have it manipulate something in the SWF file, and more.

You can add any of these types of text using the Text tool. For this exercise, you will add some static text to the Stage for decorative purposes. To add static text, follow these steps:

Open a new Flash document and name it **text_test.fla** in the Unit_5 folder.

1. With your text tool, draw a text block, and type in the words "Macromedia Flash".
2. In the Property inspector, change the following font properties:
 - Font: Your choice!
 - Size: 60
 - Color: Red



3. Change each of the Anti-alias settings to view the subtle visual changes.

Using Bitmaps in Flash 8

Setting bitmap properties

You can apply anti-aliasing to an imported bitmap to smooth the edges in the image. You can also select a compression option to reduce the bitmap file size and format the file for display on the web.

To set bitmap properties:

Select a bitmap in the Library panel. If you don't have one – import one in.

Do one of the following:

Click the properties icon at the bottom of the Library panel.

Right-click (Windows) or Control-click (Macintosh) the bitmap's icon and select Properties from the context menu.

Select Properties from the options menu in the upper right corner of the Library panel.

In the Bitmap Properties dialog box, select Allow Smoothing to smooth the edges of the bitmap with anti-aliasing.

For Compression, select one of the following options:

Photo (JPEG) compresses the image in JPEG format. To use the default compression quality specified for the imported image, select Use Document Default Quality. To specify a new quality compression setting, deselect Use Document Default Quality and enter a value between 1 and 100 in the Quality text box. (A higher setting preserves greater image integrity but yields a larger file size.)

6. Lossless (PNG/GIF) compresses the image with lossless compression, in which no data is discarded from the image.

NOTE: Use Photo compression for images with complex color or tonal variations, such as photographs or images with gradient fills. Use Lossless compression for images with simple shapes and relatively few colors.

7. **Click Test** to determine the results of the file compression. Compare the original file size to the compressed file size to determine if the selected compression setting is acceptable.
8. Click OK.

NOTE:

JPEG Quality settings that you select in the Publish Settings dialog box do not specify a quality setting for imported JPEG files. You must specify a quality setting for imported JPEG files in the Bitmap Properties dialog box.

When to enable caching

Enabling caching for a movie clip creates a surface, which has several advantages, such as helping complex vector animations to render fast. There are several scenarios in which you will want to enable caching. It might seem as though you will always want to enable caching to improve the performance of your SWF files; however, there are situations in which enabling caching does not improve performance, or even decrease it. This section describes scenarios in which caching should be used, and when to use regular movie clips.

Overall performance of cached data depends on how complex the vector data of your instances are, how much of the data you change, and whether or not you set the `opaqueBackground` property. If you are changing small regions, the difference between using a surface and using vector data could be negligible. You might want to test both scenarios with your work before you deploy the application.

When to use bitmap caching

The following are typical scenarios in which you might see significant benefits when you enable bitmap caching.

Complex background image

An application that contains a detailed and complex background image of vector data (perhaps an image where you applied the trace bitmap command, or artwork that you created in Adobe Illustrator). You might animate characters over the background, which slows the animation because the background needs to continuously regenerate the vector data. To improve performance, you can select the content, store it in a movie clip, and set the `opaqueBackground` property to true. The background is rendered as a bitmap and can be redrawn quickly, so that your animation plays much faster.

Scrolling text field

An application that displays a large amount of text in a scrolling text field. You can place the text field in a movie clip that you set as scrollable with scrolling bounds (the `scrollRect` property). This enables fast pixel scrolling for the specified instance. When a user scrolls the movie clip instance, Flash shifts the scrolled pixels up and generates the newly exposed region instead of regenerating the entire text field.

Windowing system

An application with a complex system of overlapping windows. Each window can be open or closed (for example, web browser windows). If you mark each window as a surface (set the `cacheAsBitmap` property to true), each window is isolated and cached. Users can drag the windows so that they overlap each other, and each window doesn't need to regenerate the vector content.

All of these scenarios improve the responsiveness and interactivity of the application by optimizing the vector graphics.

You can find a sample source file that shows you how bitmap caching can be applied to an instance. Find the file called `cacheBitmap.fla`, in the Samples folder on your hard disk.

- In Windows, browse to `boot drive\Program Files\Macromedia\Flash 8\Samples and Tutorials\Samples\ActionScript\CacheBitmap`.
- On the Macintosh, browse to `Macintosh HD/Applications/Macromedia Flash 8/Samples and Tutorials/Samples/ActionScript/CacheBitmap`.

You can also find a sample source file that shows you how to apply bitmap caching to scrolling text. Find the sample source file, `flashtype.fla`, in the Samples folder on your hard disk.

- In Windows, browse to `boot drive\Program Files\Macromedia\Flash 8\Samples and Tutorials\Samples\ActionScript\FlashType`.
- On the Macintosh, browse to `Macintosh HD/Applications/Macromedia Flash 8/Samples and Tutorials/Samples\ActionScript/FlashType`.

When to avoid using bitmap caching

Misusing this feature could negatively affect your SWF file. When you develop a FLA file that uses surfaces, remember the following guidelines:

- Do not overuse surfaces (movie clips with caching enabled). Each surface uses more memory than a regular movie clip, which means that you should only enable surfaces when you need to improve rendering performance.

A cached bitmap can use significantly more memory than a regular movie clip instance. For example, if the movie clip on Stage is 250 pixels by 250 pixels in size, when cached it might use 250 KB instead of 1 KB when it's a regular (uncached) movie clip instance.

- Avoid zooming into cached surfaces. If you overuse bitmap caching, a large amount of memory is consumed (see previous bullet), especially if you zoom in on the content.
- Use surfaces for movie clip instances that are largely static (nonanimating). You can drag or move the instance, but the contents of the instance should not animate or change a lot. For example, if you rotate or transform an instance, the instance changes between the surface and vector data, which is difficult to process and negatively affects your SWF file.
- If you mix surfaces with vector data, it increases the amount of processing that Flash Player (and sometimes the computer) needs to do. Group surfaces together as much as possible; for example, when you create windowing applications.

Runtime Bitmap Caching - Movie Clip and Button Symbols

Runtime bitmap caching lets you optimize playback performance by specifying that a static movie clip (for example, a background image) or button symbol be cached as a bitmap at runtime. Caching a movie clip as a bitmap prevents Flash Player from having to continually redraw the image, which provides a significant improvement in playback performance.

For example, when creating animations with a complex background, you can create a movie clip for the background. The background is rendered as a bitmap stored at the current screen depth. It can be drawn very quickly, letting the animation play both faster and more smoothly, because the background doesn't need to continually be redrawn.

Without the use of bitmap caching, the animation might play back too slowly, because the background continually would be redrawn from vector data.

Bitmap caching lets you use a movie clip and "freeze" it in place automatically. If a region changes, Flash uses vector data to update the bitmap cache. This minimizes the number of redraws that Flash Player must perform, and provides smoother, faster playback performance.

Only use runtime bitmap caching on static, complex movie clips in which the position, but not the content, of the movie clip changes on each frame in an animation. You can only observe on complex-content movie clips the playback or runtime performance improvement in using runtime bitmap caching. You will not see the performance benefit of runtime bitmap caching when creating simple movie clips.

NOTE:

You can only apply the Use Runtime Bitmap Caching check box to movie clip and button symbols.

Under the following circumstances, a movie clip will not use a bitmap (even if the Use Runtime Bitmap Caching check box is selected) and will instead render the movie clip or button symbol using vector data:

- The bitmap is too large (greater than 2880 pixels in either direction).
- The bitmap fails to allocate (producing an out of memory error).
- The parent surface uses a vector clipper (the parent is partially rotated and scrolled).

[E](#)To specify bitmap caching for a movie clip:

1. Select the movie clip or button symbol on the Stage.
2. In the symbol Property inspector, select the Use Runtime Bitmap Caching check box.

Loading Graphics Dynamically

Loading GIF, PNG and Progressive JPEGs

To load a SWF or image file, use the `loadMovie()` or `loadMovieNum()` global function, the `loadMovie()` method of the `MovieClip` class, or the `loadClip()` method of the `MovieClipLoader` class. For more information on the `loadClip()` method, see `MovieClipLoader.loadClip()` in the *ActionScript 2.0 Language Reference*.

For image files, Flash Player 8 supports the JPEG (progressive and non-progressive) image file type, GIF images (transparent and non-transparent, although only the first frame of an animated GIF will be loaded), and PNG files (transparent and non-transparent).

To load a SWF or image file into a level in Flash Player, use the `loadMovieNum()` function. To load a SWF or image file into a movie clip target, use the `loadMovie()` function or method. In either case, the loaded content replaces the content of the specified level or target movie clip.

When you load a SWF or image file into a movie clip target, the upper-left corner of the SWF file or image is placed on the registration point of the movie clip. Because this registration point is often the center of the movie clip, the loaded content might not appear centered. Also, when you load a SWF file or image to a root timeline, the upper-left corner of the image is placed on the upper-left corner of the Stage. The loaded content inherits rotation and scaling from the movie clip, but the original content of the movie clip is removed.

You can optionally send ActionScript variables with a `loadMovie()` or `loadMovieNum()` call. This is useful, for example, if the URL you're specifying in the method call is a server-side script that returns a SWF or image file according to data passed from the Flash application.

When you use the global `loadMovie()` or `loadMovieNum()` function, specify the target level or clip as a parameter. The following example loads the Flash application `contents.swf` into the movie clip instance named `image_mc`:

```
loadMovie("contents.swf", image_mc);
```

You can use `MovieClip.loadMovie()` to achieve the same result:

```
image_mc.loadMovie("contents.swf");
```

The following example loads the `image1.jpg` JPEG image into the `image_mc` movie clip instance:

```
image_mc.loadMovie("http://www.helpexamples.com/flash/images/image1.jpg");
```

Using the New 9-Slice Scaling Feature

You can use 9-slice scaling (Scale-9) to specify component-style scaling for movie clips. This lets you create movie clip symbols that scale appropriately for use as user interface components, as opposed to the type of scaling typically applied to graphics and design elements.

The movie clip is conceptually divided into nine sections with a grid-like overlay, and each of the nine areas is scaled independently. To maintain the visual integrity of the movie clip, corners are not scaled, while the remaining areas of the image are scaled (as opposed to being stretched) larger or smaller, as needed.

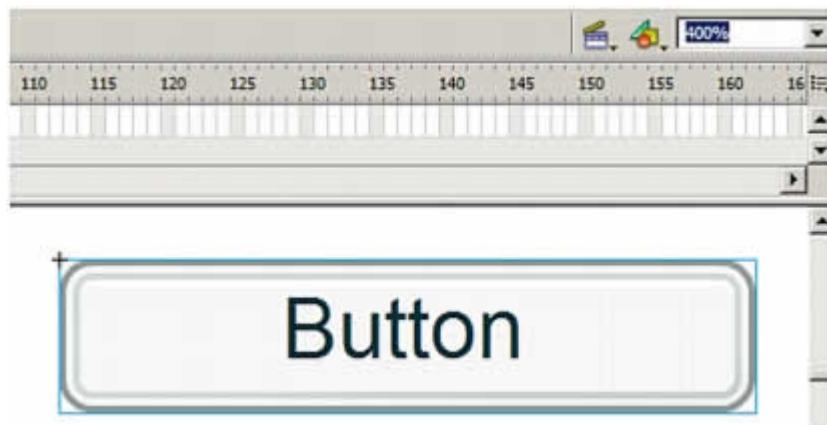
When a movie clip symbol has 9-slice scaling applied, it appears in the Library panel preview with the guides displayed. 9-slice scaling is visible only in the Test Movie window; you cannot view 9-slice scaling on the Stage.

Understanding how 9-slice scaling works

The easiest way to explain how 9-slice scaling works is to look at an example of how 9-slice scaling works in Flash.

To understand scaling in Flash:

1. Create a new Flash document and save it as **dynmask.fla**.
2. Drag a copy of the Button component to the Stage from the Components panel (Window > Components).
3. Increase the Stage's zoom level to 400% by using the Zoom tool.



By default, the Button component instance is 100 pixels wide by 22 pixels high.

4. Resize the Button component instance to 200 pixels width by 44 pixels high by using the Property inspector.

Button

You can see that even though the component resized, the Button's border and text label do not distort. The button's label remained centered and maintained its font size.

Although components of version 2 of the Macromedia Component Architecture do not use 9-slice scaling, components handle scaling in the version 2 component architecture so the outlines do not change size (as shown in the next figure).

Imagine that the button instance is sliced into 9 separate pieces, or a 3 by 3 grid, similar to a keypad on a telephone or keyboard. When you resize the button instance horizontally, only the three vertical segments in the center (numbers 2, 5, and 8 on a keypad) stretch so your content doesn't appear distorted.

If you resized the button instance vertically, only the three horizontal segments in the center (numbers 4, 5, and 6 on a keypad) would resize. The four corners of the scaling grid are not scaled at all, which allows the component to grow without looking like it is being stretched (see the following images).



TIP:

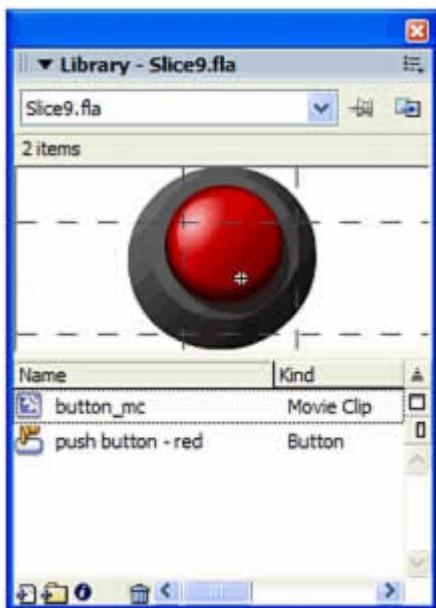
Strokes are created from the edges after the 9-slice scaling transformation, and therefore don't deform or lose detail.

You can enable slice guides for 9-slice scaling in the Flash environment within the Convert to Symbol dialog box or the Symbol Properties dialog box. The Enable guides for 9-slice scaling check box is available only if you are publishing for Flash Player 8 and the behavior is set to movie clip. The 9-slice scaling guides are not available for earlier versions of Flash or if you are creating a button or graphic symbol. 9-slice scaling can be enabled in ActionScript by setting the scale9Grid property on a movie clip instance.

Whether you created your slice guides by using the user interface or by using ActionScript, you can trace the x coordinate, y coordinate, width, and height by tracing the movie clip's scale9Grid property.

```
trace(my_mc.scale9Grid); // (x=20, y=20, w=120, h=120)
```

This snippet of code traces the value of the Rectangle object being used by the scale9Grid property. The rectangle has a x and y coordinates of 20 pixels, a width of 120 pixels and a height of 120 pixels.



Editing movie clip symbols with 9-slice scaling

By default, slice guides are placed at 25% (or 1/4) of the symbol's width and height from the edge of the symbol. When you are in symbol-editing mode, the slice guides appear with dotted lines superimposed on the symbol on the main Stage. The guides do not appear when the symbol is in edit-in-place mode. Slice guides don't snap when you drag them in the workspace.

To enable 9-slice scaling for an existing movie clip symbol:

1. With the source document open, select Window > Library to display the Library panel.
2. Select a movie clip, button, or graphic symbol in the Library panel.
3. Select Properties from the Library options menu.
4. Select the Enable Guides for 9-slice Scaling check box.

The slice guides are superimposed on the symbol on the Stage.

To edit a movie clip symbol using 9-slice scaling:

1. Enter symbol-editing mode by doing one of the following:
 - Select an instance of the symbol on the Stage and right-click (Windows) or Control-click (Macintosh), and select Edit from the context menu.
 - Select the symbol in the Library and right-click (Windows) or Control-click (Macintosh), and select Edit from the context menu.
 - Double-click the symbol in the Library.

The slice-9 guides appear.

2. Move the pointer over any of the four guides in the workspace to change the pointer to the horizontal or vertical guide pointers that indicate that a drag operation will move the position of the guide. Drag and release the pointer.

The new position of the guide is updated in the library preview for the symbol.

Using ActionScript to Modify Bitmaps

Creating Bitmaps with the `BitmapData` Class

The `BitmapData` class lets you create arbitrarily sized transparent or opaque bitmap images, then manipulate them in various ways at runtime. When you manipulate a `BitmapData` instance directly by using ActionScript, you can create very complex images without incurring the overhead of constantly redrawing the content from vector data in Flash Player. The methods of the `BitmapData` class support a variety of effects that are not available through the Filters tab in the Flash workspace.

A `BitmapData` object contains an array of pixel data. This data either can represent a fully opaque bitmap or a transparent bitmap that contains alpha channel data. Either type of `BitmapData` object is stored as a buffer of 32-bit integers. Each 32-bit integer determines the properties of a single pixel in the bitmap. Each 32-bit integer is a combination of four 8-bit *channel* values (from 0 to 255) that describe the alpha transparency and the red, green, and blue (ARGB) values of the pixel.

You can find a sample source file that uses the `BitmapData` class to manipulate an image. Find the file called `BitmapData.fla` in the Samples folder on your hard disk.

- In Windows, browse to *boot drive\Program Files\Macromedia\Flash 8\Samples and Tutorials\Samples\ActionScript\BitmapData*.
- On the Macintosh, browse to *Macintosh HD/Applications/Macromedia Flash 8/Samples and Tutorials/Samples/ActionScript/BitmapData*.

The following procedure dynamically loads a JPEG image onto the Stage, and uses the `BitmapData` class to create a noise effect, similar to static on a television. The noise effect is redrawn with a random pattern every 100 milliseconds (1/10 of a second). Moving the mouse pointer along the x-axis and y-axis affects how much static is drawn at every interval.

To create a noise effect with the `BitmapData` class:

1. Create a new Flash document and save it as **noise.fla**.
2. Add the following ActionScript to Frame 1 of the Timeline:

```
import flash.display.BitmapData;
this.createTextField("status_txt", 90, 0, 0, 100, 20);
status_txt.selectable = false;
status_txt.background = 0xFFFFFFFF;
status_txt.autoSize = "left";
function onMouseMove() {
    status_txt._x = _xmouse;
    status_txt._y = _ymouse-20;
    updateAfterEvent();
}
this.createEmptyMovieClip("img_mc", 10);
img_mc.loadMovie("http://www.helpexamples.com/flash/images/image1.jpg");
var noiseBmp:BitmapData = new BitmapData(Stage.width,
Stage.height, true);
this.attachBitmap(noiseBmp, 20);
setInterval(updateNoise, 100);
var grayScale:Boolean = true;
function updateNoise():Void {
    var low:Number = 30 * _xmouse / Stage.width;
    var high:Number = 200 * _ymouse / Stage.height;
    status_txt.text = "low:" + Math.round(low) + ", high:" +
Math.round(high);
    noiseBmp.noise(Math.round(Math.random() * 100000), low, high,
8, true);
}
```

This code creates a text field with the instance name `status_txt`, which follows the mouse pointer and displays the current values for the high and low parameters for the `noise()` method. The `setInterval()` function changes the noise effect, which is updated every 100 milliseconds (1/10 of a second), by continuously calling the `updateNoise()` function. The high and low parameters for the `noise()` method are determined by calculating the pointer's current position on the Stage.

3. Select **Control > Test Movie** to test the document.

Moving the mouse pointer along the x-axis affects the low parameter; moving the mouse pointer along the y-axis affects the high parameter.

The `BitmapData` class also lets you distort a dynamically loaded image by using a combination of a `perlinNoise()` method effect and a displacement map filter. The following procedure shows this.

Using the Displacement Map Filter

The `DisplacementMapFilter` class uses the pixel values from the specified `BitmapData` object (called the *displacement map image*) to perform a displacement of an instance that's on the Stage, such as a movie clip instance or a bitmap data instance. You can use this filter to achieve a warped or mottled effect on a specified instance.

This filter is only available by using ActionScript.

The following procedure loads a JPEG image and applies a displacement map filter to it, which causes the image to look distorted. Whenever the user moves the mouse, the displacement map is regenerated.

To distort an image with the displacement map filter:

1. Create a new Flash document and save it as **displacement.fla**.
2. Add the following ActionScript to Frame 1 of the Timeline:

```
import flash.filters.DisplacementMapFilter;
import flash.geom.Point;
import flash.display.BitmapData;

var perlinBmp:BitmapData;
var displacementMap:DisplacementMapFilter;
var mclListener:Object = new Object();
mclListener.onLoadInit = function(target_mc:MovieClip):Void {
    target_mc._x = (Stage.width - target_mc._width) / 2;
    target_mc._y = (Stage.height - target_mc._height) / 2;
    perlinBmp = new BitmapData(target_mc._width,
    target_mc._height);
    perlinBmp.perlinNoise(target_mc._width, target_mc._height,
    10, Math.round(Math.random() * 100000), false, true, 1, false);
    displacementMap = new DisplacementMapFilter(perlinBmp, new
    Point(0, 0), 1, 1, 100, 100, "color");
    shapeClip.filters = [displacementMap];
};
var shapeClip:MovieClip = this.createEmptyMovieClip("shapeClip",
1);
shapeClip.createEmptyMovieClip("holderClip", 1);
var imageLoader:MovieClipLoader = new MovieClipLoader();
imageLoader.addListener(mclListener);
imageLoader.loadClip("http://www.helpexamples.com/flash/images/im
age1.jpg", shapeClip.holderClip);

var mouseListener:Object = new Object();
mouseListener.onMouseMove = function():Void {
    perlinBmp.perlinNoise(shapeClip._width, shapeClip._height,
    10, Math.round(Math.random() * 100000), false, true, 1, false);
```

```
shapeClip.filters = [displacementMap];  
};  
Mouse.addListener(mouseListener);
```

This code loads a JPEG image and places it on the Stage. After the image is completely loaded, this code creates a `BitmapData` instance and uses the `perlinNoise()` method to fill it with randomly placed pixels. The `BitmapData` instance passes to the displacement map filter, which is applied to the image and causes the image to look distorted.

3. Select Control > Test Movie to test the document.



Move your mouse pointer around the Stage to re-create a displacement map by calling the `perlinNoise()` method, which changes the appearance of the JPEG image.

Using the Convolution Filter

The `ConvolutionFilter` class applies a matrix convolution filter effect. A convolution combines pixels in a source image that you specify with neighboring pixels to produce an image. You can achieve a wide variety of imaging operations by using the convolution filter, which includes blurring, edge detection, sharpening, embossing, and beveling effects.

NOTE: You can apply this filter on bitmaps and movie clip instances

A matrix convolution is based on an $n \times m$ matrix, which describes how a given pixel value in the input image is combined with its neighboring pixel values to produce a resulting pixel value. Each resulting pixel is determined by applying the matrix to the corresponding source pixel and its neighboring pixels.

This filter is only available by using ActionScript.

The following procedure applies the convolution filter to a dynamically loaded JPEG image.

To use the convolution filter to modify an image's color:

1. Create a new Flash document and save it as **convolution.fla**.
2. Add the following ActionScript to Frame 1 of the Timeline:

```
import flash.filters.ConvolutionFilter;
import flash.display.BitmapData;

this.createEmptyMovieClip("shape_mc", 1);
shape_mc.createEmptyMovieClip("holder_mc", 1);
var imageLoader:MovieClipLoader = new MovieClipLoader();
imageLoader.loadClip("http://www.helpexamples.com/flash/images/ima
ge1.jpg", shape_mc.holder_mc);
var matrixArr:Array = [1, 4, 6, 4, 1, 4, 16, 24, 16, 4, 16, 6,
24, 36, 24, 6, 4, 16, 24, 16, 4, 1, 4, 6, 4, 1];
var convolution:ConvolutionFilter = new ConvolutionFilter(5, 5,
matrixArr);
shape_mc.filters = [convolution];

var mouseListener:Object = new Object();
mouseListener.onMouseMove = function():Void {
    convolution.divisor = (_xmouse / Stage.width) * 271;
    convolution.bias = (_ymouse / Stage.height) * 100;
    shape_mc.filters = [convolution];
};
Mouse.addListener(mouseListener);
```

The preceding code is separated into three sections. The first section imports two classes: ConvolutionFilter and BitmapData. The second section creates a nested movie clip and uses a movie clip loader object to load an image into the nested movie clip. A convolution filter object is created and applied to the shape_mc movie clip. The final section of code defines a mouse listener object that modifies the convolution filter's divisor and bias properties based on the current position of the mouse pointer and reapplies the convolution filter to the shape_mc movie clip.

3. Select **Control > Test Movie** to test the Flash document.

Moving the mouse pointer along the x-axis of the Stage modifies the filter's divisor, whereas moving the mouse pointer along the y-axis of the Stage modifies the filter's bias.

Lab

- Create some text, convert it into a symbol and apply a filter.
- Animate the text and try the new ease in/ease out controls.

New Features Summary

- Blends
- Programming Blends with ActionScript
- Filters
- Programming Filters with ActionScript
- Importing From Fireworks
- Bitmaps in Flash 8
- Using the New Easing In/Easing Out Controls

Unit 7: Using Video in Flash 8

Unit Objectives:

After this unit, you will be able to:

- Import Video with the New Video Encoder
- Work with Alpha Channels
- Batch Process Video
- Use the New On2 VP6 Video Codec
- Embed Cue Points
- Build a Video Player in Flash 8

Unit Topics:

- Video Improvements
- Importing Video
- Working with Alpha Channels
- The New On2 VP6 Video Codec
- Embedding Cue Points
- Building a Flash Video Player

Video improvements

The list of improvements to video includes:

- A new video codec (On2 VP6) for better quality video and smaller video file size
- Improved video import wizard
- Stand alone video encoder
- Alpha channel support
- Embedded cue points for triggering events from specific points in a video

Comparing the On2 VP6 and Sorenson Spark video codecs

The On2 VP6 codec is the default video codec to use when encoding FLV content for use with Flash Player 8. The On2 VP6 codec provides:

- Higher quality video when compared to the Sorenson Spark codec encoded at the same data rate
- Support for the use of an alpha channel to composite video

In order to support better quality video at the same data rate, the On2 VP6 codec requires noticeably slower to encode and requires more processor power on the client computer to decode and play back. For this reason, you should carefully consider the lowest common denominator of computer you intend your viewing audience to use when accessing your Flash Video content.

If you anticipate a large user base that is using older computers, it may be advisable to encode your FLV files using the Sorenson Spark codec.

About the On2 VP6 and Sorenson Spark video codecs

By default, Flash Video Encoder exports encoded video using the On2 VP6 video codec for use with Flash Player 8, and the Sorenson Spark codec for use with Flash Player 7. A *codec* is a compression/decompression algorithm that controls how video files are compressed during encoding, and decompressed during playback. The On2 video codec is the preferred video codec to use when creating Flash content that uses video. On2 provides the best combination of video quality while maintaining a small file size.

If your Flash content dynamically loads Flash video (using either progressive download or Flash Communication Server), you can use On2 VP6 video without having to republish your SWF for Flash Player 8, as long as users use Flash Player 8 to view your content. By streaming or downloading On2 VP6 video into Flash SWF versions 6 or 7, and playing the content using Flash Player 8, you avoid having to recreate your SWF files for use with Flash Player 8.

About encoding video

Flash provides several video encoding solutions that let you encode your video clips into the FLV format.

Flash Video Import wizard

The Flash Video Import wizard lets you encode video clips into the Flash Video (FLV) format when you import them. However, the Video Import wizard has limitations in that you can only encode one video clip at a time, and the process of encoding can be both time- and computing-intensive.

For users who work extensively with video-based content, Flash Professional 8 includes the Flash 8 Video Encoder and the QuickTime Exporter.

Flash 8 Video Encoder lets you batch process video clips, allowing you to encode several clips at a time without having to interrupt your workflow. In addition to selecting encoding options for video and audio content, the Flash 8 Video Encoder also lets you embed cue points into video clips you encode, and edit the video using crop-and-trim controls.

For more information, see the online help included with the Flash 8 Video Encoder application.

FLV QuickTime Export plug-in If you have Macromedia Flash Professional 8 and QuickTime 6.1.1 installed on your computer, you can use the FLV QuickTime Export plug-in to export FLV files from supported video-editing applications. You can then import these FLV files directly into Flash to use in your Flash documents.

The following video-editing applications are supported by the FLV Export plug-in:

- Adobe After Effects (Windows and Macintosh)
- Apple FinalCut Pro (Macintosh)
- Apple QuickTime Pro (Windows and Macintosh)
- Avid Xpress DV (Windows and Macintosh)

Using the FLV QuickTime Export plug-in to export FLV files from either Flash 8 Video Encoder or video-editing applications significantly streamlines the process of working with FLV files in your Flash documents. With the FLV Export plug-in, you can select encoding options for video and audio content as you export, including frame rate, bit rate, quality, and other options. You can import FLV files directly into Flash without needing to re-encode the video after import.

Tips for creating Flash video

How you compress your video is largely determined by the content of the video. A video clip of a talking head with little action and only short bursts of moderate motion compresses differently from footage of a soccer match. Follow these guidelines to deliver the best possible Flash video:

Whenever possible, always encode a file from its uncompressed form If you convert a precompressed digital video format into the FLV format, the previous encoder can introduce video noise. The first compressor has already performed its encoding algorithm on the video and has already reduced its quality, frame size, and rate. It may have also introduced some of its own digital artifacts or noise. This additional noise affects the FLV encoding process and may require a higher data rate to play back a good-quality file.

Strive for simplicity Avoid elaborate transitions--they don't compress well and can make your final compressed video look "chunky" during the change. Hard cuts (as opposed to cross-dissolves) are usually best. Video sequences that show an object zooming from behind the first track, doing a "page turn," or wrapping around a ball and then flying off the screen can be eye-catching, but they usually don't compress well and should be used sparingly.

Know your audience data rate When you deliver video over the Internet, you should produce files at lower data rates. Users with fast Internet connections can view the files with little or no wait, but dial-up users must wait for files to download. It is best to make the clips short to keep the download times within acceptable limits for dial-up users.

Select the proper frame rate Frame rate indicates how many frames play each second (fps). If you have a higher data rate clip, a lower frame rate can improve playback on lower-end computers. For example, if you are compressing a talking head clip with little motion, cutting the frame rate in half will probably save you only 20 percent of the data rate. However, if you are compressing high-motion video, reducing the frame rate has a much greater effect on the data rate.

Because video looks much better at native frame rates, Macromedia recommends leaving the frame rate high if your delivery channels and playback platforms allow it. However, if you need to reduce the frame rate, the best results come from dividing the frame rate by whole numbers.

NOTE:

If you intend to embed video clips within the SWF file, the frame rate of the video clip must be the same as that used by the SWF. Using the Advanced Video Encoding settings in the Video Import wizard, you can encode video using the same frame rate as the FLA file.

Select a frame size that fits your data rate As with the frame rate, the frame size for your document is important for producing high-quality video. At a given data rate (connection speed), increasing the frame size results in decreased video quality. When you select the frame size for your document, you must consider frame rate, source material, and personal preferences. The following list of common frame sizes (in pixels) can be used as a guideline. You can experiment to find the best setting for your project.

- Modem: 160 x 120
- Dual ISDN: 192 x 144
- T1/DSL/cable: 320 x 240

Know progressive download times You should know how long it is going to take to download your video. While your video clip downloads, you might want to have other content that appears and "disguises" the download. For short clips, you can use the following formula: Pause = download time - play time + 10% of play time. For example, if your clip is 30 seconds long and it takes one minute to download, you should give your clip a 33-second buffer (60 seconds - 30 seconds + 3 seconds = 33 seconds).

Use clean video The higher the quality of the original, the better the final result. Although frame rates and sizes of Internet video are usually smaller than those of television, computer monitors have much better color fidelity, saturation, sharpness, and resolution than conventional televisions. Even with a small window, image quality can be more important for digital video than for standard analog television. Artifacts and noise that would hardly be noticeable on TV can be quite obvious on a computer screen.

Remove noise and interlace After you capture your video content, you might need to remove noise and interlacing.

Follow the same guidelines for audio The same considerations exist for audio production as for video production. To achieve good audio compression, you must begin with clean audio. If you are encoding material from a CD, try to record the file using direct digital transfer instead of through the analog input of your sound card. The sound card introduces an unnecessary digital-to-analog and analog-to-digital conversion that can create noise in your source audio. Direct digital transfer tools are available for Windows and Macintosh platforms. If you must record from an analog source, you should use the highest quality sound card available.

Flash 8 Video Encoder

A new video encoder application is included with Flash Professional 8. It is a separate application that provides an easy way to convert video files into the Flash Video (FLV) format. The application also allows you to perform batch processing of video files.



macromedia®

FLASH®8 Video Encoder

Selecting a video encoding profile

Flash provides several preconfigured encoding profiles that you can use to encode your video. In the Encoding panel, you can select an encoding profile to determine the level of compression to be applied to the video clip.

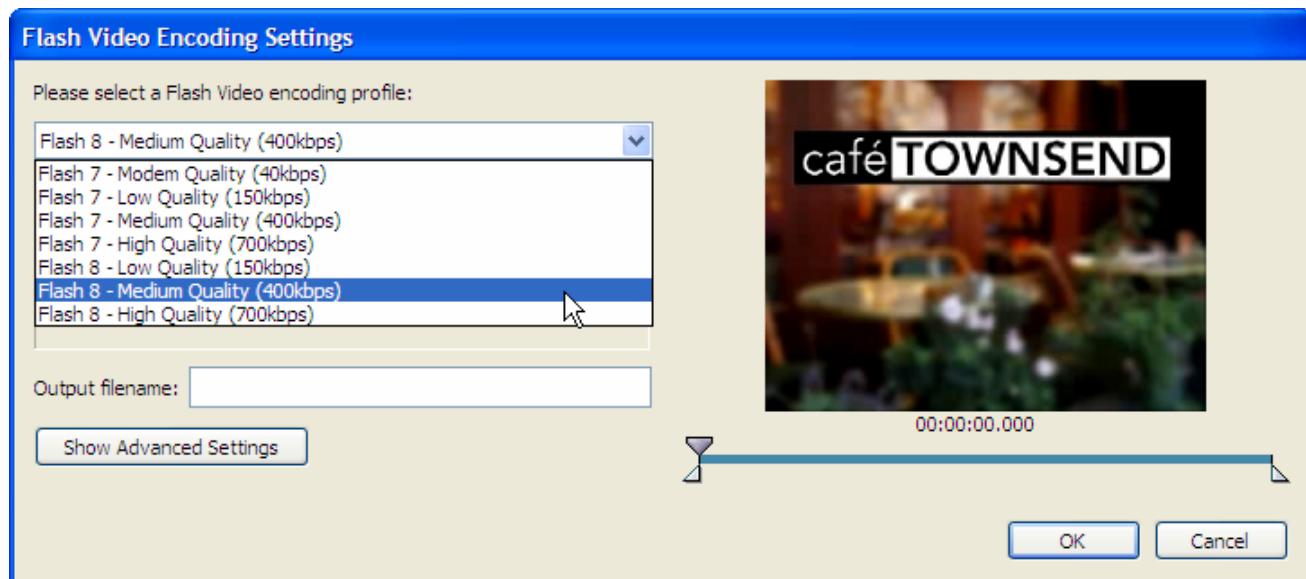
The encoding profiles are based on the Flash Player version you intend to publish content for, and the data rate at which you want your video content to be encoded. If you choose an encoding profile using Flash Player 8, the On2 VP6 video codec is used to encode the video. If choose an encoding profile using Flash Player 7, the Sorenson Spark video codec is used to encode the video. For more information on supported video codec's and Flash Player compatibility, see *About the On2 VP6 and Sorenson Spark video codecs*.

To select a video encoding profile:

In the Encoding panel, select an encoding profile from the Flash Video encoding profile pop-up menu.

Verify that the encoding profile you've selected is appropriate for your intended application.

The text box below the Flash Video encoding profile pop-up menu displays the Flash Player version, video codec, video bitrate, and audio encoding information for the selected encoding profile.



Do one of the following:

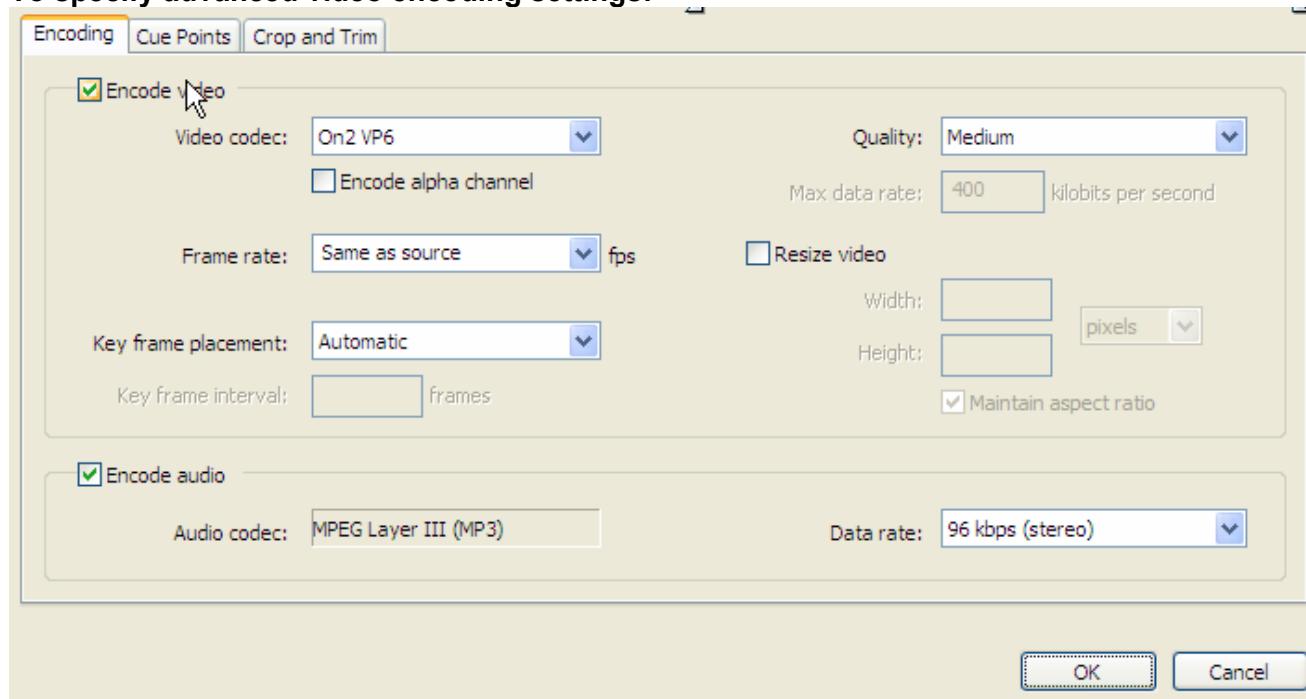
- Click Show Advanced Settings to further adjust the encoding settings, or to modify the video clip's size or playback length using the crop-and-trim controls.
- Click Continue to encode the video.

Specifying advanced encoding settings

The Video Import wizard and the FLV QuickTime Export plug-in let you create your own video encoding settings to fine-tune the quality and download size of video clips.

For information on importing video using the Video Import wizard, see Importing video using the Video Import wizard. For information on the FLV QuickTime Export plug-in, see About encoding video.

To specify advanced video encoding settings:



The Encode Video check box should be selected by default. If it is not, select it to activate the advanced encoding options.

Select a video codec with which to encode your content from the Video codec pop-up menu. If you are authoring for Flash Player 6 or 7, choose the Sorenson Spark codec; if you are authoring for Flash Player 8, choose the On2 VP6 codec.

Select a frame rate.

By default, Flash Video Encoder uses the same frame rate as the source video. Macromedia recommends using the default frame rate unless you are very experienced with video encoding, and have a specific application that requires modifying the source video's frame rate. If you need to alter the frame rate, be sure you understand how modifying the frame rate affects the video quality.

NOTE: If you intend to embed video clips within the SWF file, the frame rate of the video clip must be the same as that used by the SWF. Using the Advanced Video Encoding settings in the Video Import wizard, you can encode video using the same frame rate as the FLA file. For more information, see [About embedding video in a SWF file](#).

Select the keyframe placement for the video.

Keyframes are video frames that contain complete data. For example, if you specify a keyframe interval of 30, Flash Video Encoder encodes a complete frame every 30 frames in the video clip. For frames between keyframe intervals, Flash stores only the data that changes from the preceding frame.

By default, Flash Video Encoder places a keyframe every two seconds of playback time. For example, if the video you're encoding has a frame rate of 30 fps, a keyframe will be inserted every 60 frames. In general, the default keyframe value provides a reasonable level of control when seeking within a video clip. If you need to select a custom keyframe placement value, be aware that the smaller the keyframe interval, the larger the file size.

Specify the quality of the video from the Quality pop-up menu.

The quality setting determines the data rate (or bit rate) of the encoded video. The higher the data rate, the better the quality of the encoded video clip. To specify a quality setting, do one of the following:

Select a preset quality setting (Low, Medium, or High) to automatically select a Data Rate value. When you select Low, Medium, and High from this menu, the Maximum Data Rate text box updates to reflect the specified value.

Select Custom and enter a value, in Kilobits/Sec, in the Maximum Data Rate text box.

NOTE:

If you find that the preset quality settings are not working with your particular source footage, try specifying a custom maximum data rate. For more information on data rate and how it affects video quality, see [About encoding video](#).

To resize the video clip, do the following:

- Select the Resize video check box.
- (Optional) Select the Maintain Aspect Ratio check box to keep the aspect ratio the same as the original video clip.

NOTE:

If you resize a video clip's frame size, and do not select the Maintain Aspect Ratio check box, the video may become distorted.

- Specify values for Width and Height. You can specify a frame size in pixels or as a percentage of the original image size.
- If you are done specifying advanced encoding settings, click Next to move to the next panel of the Video Import wizard, or select either the Cue Point or the Crop and Trim tab to further modify the video's encoding settings.

Specifying advanced audio encoding settings

The audio encoding settings portion of the dialog box is automatically selected when an audio-only file is added to the encoding list (this portion of the dialog box is disabled when encoding video content without a combined audio track).

NOTE:

You can select an encoding profile from the Flash Video encoding profile pop-up menu, which will set a corresponding profile for the audio-only encoding format.

To specify custom audio encoding settings:

If the Encode Audio check box is not selected, select it to activate the advanced audio encoding options.

The Encode Audio check box should be selected by default. The default audio codec is MP3.

NOTE:

If the source video file has no audio track, or if you are encoding MPEG-1 files on a Macintosh, the audio encoding settings portion of the dialog box is disabled.

Select a data rate from the Data rate pop-up menu.

The data rate is the bit rate of the MP3 audio stream. Better quality audio tracks, such as music and significant background noise, require a higher bit rate. Simple dialogue, such as with a "talking head" shot, can be compressed to a much higher degree. Higher bit rate settings (encoded at 80 Kbps or higher) are encoded in stereo, while lower bit rate settings (encoded at 64 Kbps or lower) are encoded in mono.

Click Next to move to the next panel of the Video Import wizard, or select another of the Advanced Settings tabs to further modify the video clip.

Embedding Cue Points

Cue points cause the video playback to trigger other actions within the presentation. For example, you can create a Flash presentation that has video playing in one area of the screen while text and graphics appear in another area. A cue point placed in the video triggers an update to the text and graphic, letting them remain relevant to the content of the video.

Each cue point consists of a name and the time at which it occurs. You specify cue point times in hour:minute:second:millisecond format; the default frame rate is 30 frames per second (fps). You can specify cue point times with any frame rate, and also express them in milliseconds rather than frame numbers.

To define and embed cue points, you must either use Flash Video Encoder or import a video clip using the Video Import wizard. For information on importing video using the Video Import wizard, see [Importing video using the Video Import wizard](#).

To define and embed cue points in a video clip:

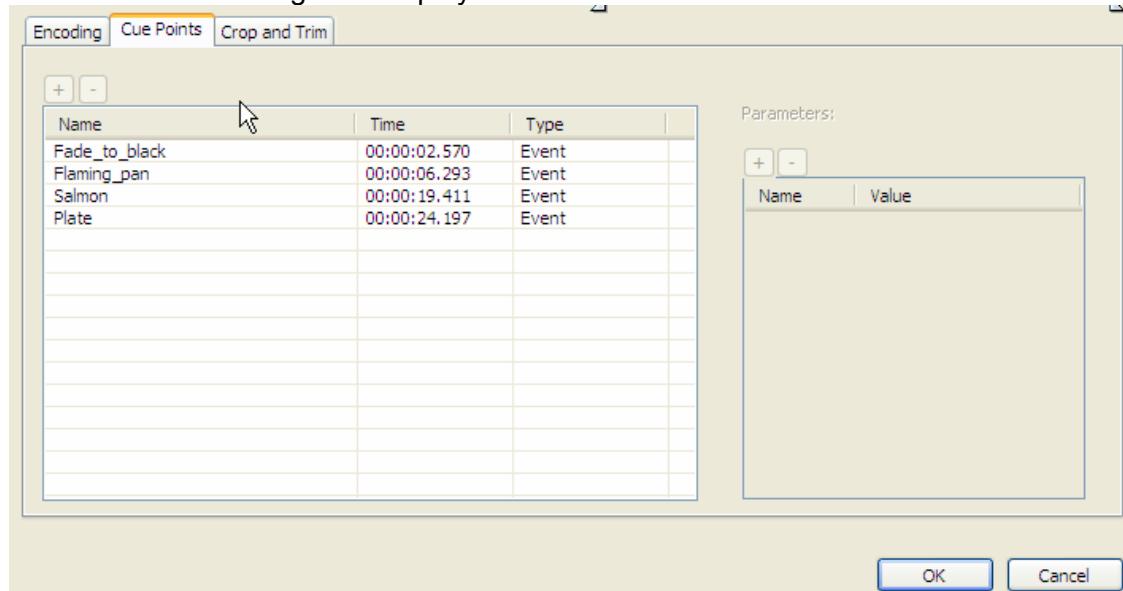
From the Encoding panel of the Video Import wizard, click Show Advanced Settings.

Select a predefined encoding profile from the Flash Video encoding profile pop-up menu, or create your own custom encoding profile using the encoding options in the Encoding tab.

For information on specifying the custom video encoding settings, see [Encoding video](#).

Click the Cue Points tab.

The Cue Points settings are displayed.



Use the playback head to locate a specific frame (point in the video) where you want to embed a cue point. For greater precision, select the playback head, and use the left and right arrow keys to locate specific points within the video.

To locate a specific frame, use the pointer to move the playback head to the point in the video where you want to embed a cue point. The video preview window lets you visually identify points in the video at which to insert a cue point. You can also use the elapsed time counter (located beneath the video preview window) to locate specific points in time at which to embed cue points.

When the playback head is positioned on a frame where you want to embed a cue point, click the Add Cue Point button.

Flash Video Encoder embeds a cue point on that frame of the video, and populates the cue point list

with a placeholder for the name of the new cue point, and the elapsed time and video frame at which the cue point is located (this is the time during playback when the event will be triggered). Flash Video Encoder also displays a pop-up menu that lets you select the type of cue point to embed.

A cue point marker is displayed on the slider control at the point where the cue point was embedded. You can use the cue point marker to adjust the placement of the cue point.

Specify the type of cue point you want to embed, an event cue point or a navigation cue point: Event cue points are used to trigger ActionScript methods when the cue point is reached, and let you synchronize the video playback to other events within the Flash presentation.

Navigation cue points are used for navigation and seeking, and to trigger ActionScript methods when the cue point is reached. Embedding a navigation cue point inserts a keyframe at that point in the video clip.

Enter parameters for the selected cue point.

Parameters are a set of key-value pairs that you can add to the cue point. The parameters are passed to the cue point event handler as members of the single parameter object.

Video alpha channel support You can now use an alpha channel with video objects, allowing you to create transparency effects.

To use a video file to create the alpha effect in Flash, the video file must contain an alpha channel.

In general, the video file must meet the following requirements:

- 32-bit video with million+ colors
- Includes a color-key background
- Is encoded to enable the alpha channel

Building a Video Player

This tutorial guides you through the process of creating a simple video player by using some of the authoring features in Macromedia Flash 8. After it is created, you can deploy the video player in a web page.

In this tutorial, you will complete the following tasks:

- Review your task
- Examine the completed application
- Create a working folder
- Encode a video file
- Create a new Flash document
- Add a media component
- Publish your document

Review your task

In this tutorial, you will create a type of application known as a flexible messaging area, or FMA. An FMA is a common type of Flash application used for displaying content that conveys some kind of informational or marketing message to the audience. In this case, the FMA displays photographs of items from the restaurant menu. At Macromedia's website, an FMA is used to display information about new software products and other advertising messages. These are called flexible messaging areas because they usually occupy an area of the web page that is set aside for content that can change depending on the needs of the business or website. For example, if Café Townsend has a special event planned, the FMA could change to display the details of that event instead of the restaurant's menu items.

In this tutorial, after examining an existing Flash document, you'll begin by creating a new Flash document and end by publishing the application for web playback. The tutorial should take approximately 20 to 30 minutes to complete.

Run the completed application

To better understand the type of application you'll create as you work through this tutorial, you can look at a completed FLA file version of the application in the Flash authoring tool. FLA files are the files you work on in Flash. You can also play the SWF version of the file in Flash Player. The SWF version is the version of the file that you would publish in a web page.

To play the SWF version of the file in Flash Player:

1. In Flash, select **File > Open**.
2. Use one of the following paths to browse to the completed file:

- a. In Windows, browse to *Unit_7\cafe_townsend\completed files\flash* and double-click *video_pod_finished.swf*.
3. The completed application runs in Flash Player. Click the playback controls at the bottom of the window to watch the video.
4. After viewing the application, close the Flash Player window.

Open the authoring document

It's helpful to look at the completed authoring FLA file to see how the author designed the application.

To view the authoring version of the file in Flash:

1. In Flash, select **File > Open**.
2. Browse to the authoring document using one of the following paths:
 - o In Windows, browse to *Unit_7\cafe_townsend\completed files\flash* and double-click *video_pod_finished.fla*.
 - o On the Macintosh, browse to *Macintosh Unit_7/cafe_townsend/completed_files/flash* and double-click *video_pod_finished.fla*.

You now see the completed tutorial application in the Flash authoring environment.

3. After you have the file open, you can explore the Stage, Library panel, and the Timeline.
 - o On the Stage you will see a FLVPlayback component, which you will learn more about later in this tutorial.
 - o In the Library panel, you will see a list of symbols, or reusable assets, that the document uses.
 - o In the Timeline, you will see a representation of how and when those symbols appear on the Stage.

You will learn more about the role of each of these parts of Flash as you complete the tutorial.

Create a working folder

Before you begin, you must create a working folder that includes the sample files used in the tutorials in this guide, *Getting Started with Flash*. This task consists of creating the working folder on your hard disk and copying the sample files from the Flash application folder to the working folder.

1. Create a new folder called **local_sites** in the **Unit 7 folder**. For example, create a folder called **local_sites** in either of the following locations:
 - 2. Locate the **cafe_townsend** folder in the Flash application folder on your hard disk. If you installed Flash to its default location, the path to the folders is as follows:

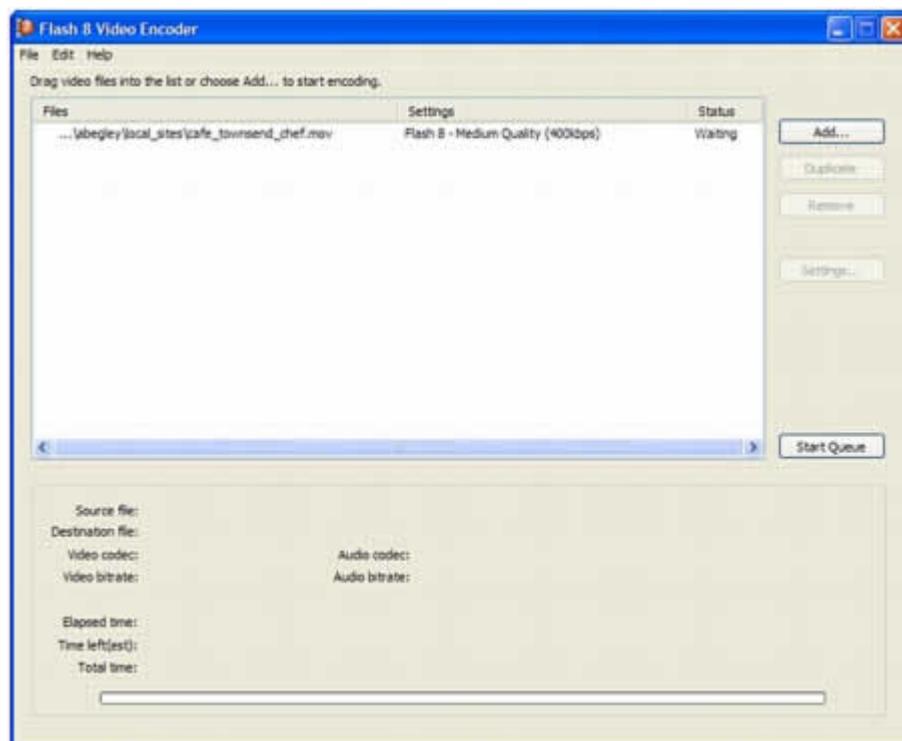
3. Copy the **cafe_townsend** folder to your new folder.

Encode a video file

The first thing you will do is convert a QuickTime video file (MOV) to a Flash video file (FLV).

To convert the video file:

1. Start the Flash Video Encoder application.
2. Drag the **cafe_townsend_chef.mov** file from the **cafe_townsend** folder onto the Flash Video Encoder application window.



The Flash Video Encoder window

3. Click **Settings**.
4. In the Flash Video Encoding Settings dialog box, select **Flash 8 - Medium Quality (400kbps)** from the Flash Video Encoding Profile menu. This setting is the default.

This action applies only a small amount of compression to the file.



5. Click **OK** to close the Settings dialog box.

6. Click **Start Queue**.

The Flash Video Encoder converts the file and saves it in the same folder as the original `cafe_townsend_chef.mov` file.

You are now ready to use the FLV file in a Flash document.

7. Close the Flash Video Encoder application.

Create a new Flash document

Now it's time to create your own Flash document.

Open a new document

Now you're ready to create your own version of the FMA.

To create a new document:

1. Start Flash.
2. Select **File > New**.
3. In the New Document dialog box, select Flash Document and then click **OK**.
4. Select **File > Save**.
5. Name the file **video_pod.fla** and save the file in the `cafe_townsend` folder you copied to your hard disk.

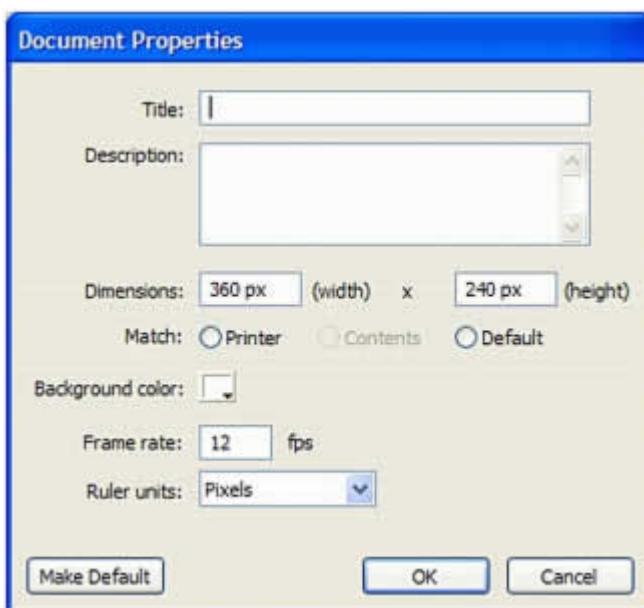
Define document properties

Configuring document properties is a common first step in authoring. You can change the document properties at any time, but it is helpful to make certain decisions, such as the Stage size and background color, at the beginning of the process.

Document properties are properties that affect the entire Flash document. You can use the Property inspector to specify these settings.

To define document properties:

1. If the Property inspector isn't open, select Window > Properties > Properties.
2. In the Property inspector, click Size.
3. In the Document Properties dialog box, enter the following values:
 - o Width: **360**
 - o Height: **240**
4. Click OK.



5. Save your work.

Add a media component

Next, you will add a media playback component to the Stage. This component will contain the video display and provide playback controls.

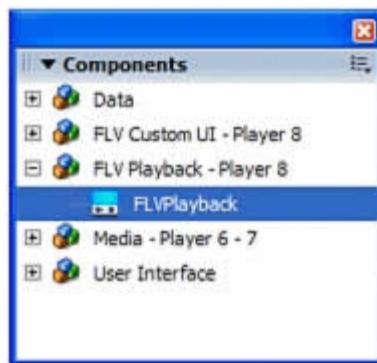
About components

Components in Flash are prebuilt objects that you can use in your own Flash documents. Most components are user interface elements such as buttons, menus, and so on. Some are not meant to be seen on the Stage and are used instead to perform data-handling functions. By using a component, you can avoid the need to build complex user interface elements yourself. Components are implemented in Flash as movie clips with their own internal ActionScript code.

You use the Components panel to set the properties of the component. This tells Flash how you want the component to behave. For the FLVPlayback component, you will tell Flash the location of the FLV file you want to play, and how you want the playback controls to appear.

To add a media component:

1. Select **Window > Components** to open the Components panel.
2. In the Components panel, click the **plus sign (+)** next to the **FLV Playback - Player 8** category.



3. Drag an FLVPlayback component to the Stage.

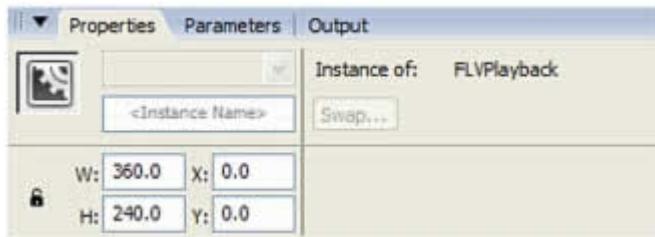
The new component remains selected on the Stage by default.



4. With the new component still selected on the Stage, enter the following values in the Property inspector:
 - o W: **360**
 - o H: **240**
 - o X: **0**

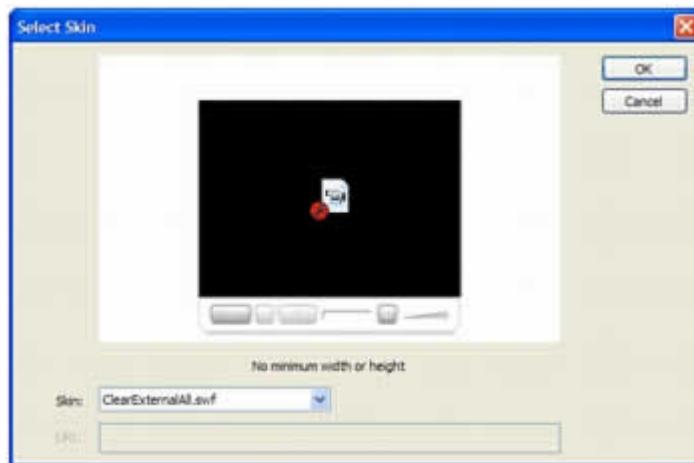
- Y: 0

This makes the component the same size as the Stage and centers it on the Stage.



The Property inspector with the proper values entered

5. With the new component still selected on the Stage, select Window > Component Inspector to open the Component inspector.
6. In the Component inspector's Parameters tab, click the contentPath parameter.
7. Click the magnifying glass icon that appears next to it.



8. In the Content Path dialog box, enter **cafe_townsend_chef.flv** and click OK.
9. Still in the Component inspector, set the autoRewind parameter to false.
10. Save your work.
11. To test your document, select Control >Test Movie.

The document plays in the Test Movie window. You can control the playback of the video file with the controls at the bottom of the Stage.

Publish your document

The next step is to publish your FLA file as a SWF file that can be displayed in a web browser.

To publish your Flash document:

1. Select **File > Publish**.

Flash saves a SWF version of your file and a simple HTML file in the `cafe_townsend` folder where you saved the `video_pod.fla` file. Flash also saves a SWF file called `ClearOverPlaySeekMute.swf` that contains the graphics for the video controller overlay that appears on top of the video. This SWF file must be located in the same folder as the `video_pod.swf` file in order for the video controls to appear when you play the SWF file.

2. In your web browser, select **File > Open**.
3. Navigate to the `cafe_townsend` folder and open the `video_pod.html` file.

The HTML file opens in the web browser and displays your Flash document.

The next steps

Now that you have completed the Flash video player, you would normally insert the SWF file into a real-world web page.

This illustration demonstrates what a web page containing the `video_pod.swf` file might look like.



The `video_pod.swf` file inserted in a web page

Lab

- Examine the video in the Unit_7/video_spin folder. There's a video with an alpha channel.

New Feature Summary

- The New On2 VP6 Video Codec
- Alpha Channels with video
- Embedding Cue Points

Unit 8: Mobile Phone Workflow

Unit Objectives:

After this unit, you will be able to:

- Describe the new mobile phone features in Flash 8
- Set up a document for Flash Lite delivery
- Build an application with common mobile device functionality
- Use the Flash Lite emulator

Unit Topics:

- What's New in Flash Lite
- Workflow Authoring for Mobile Devices
- Review the Café Application
- Creating the Café Application
- Previewing the Café Application
- Previewing a Mobile Video Game

What's New in Flash Lite Authoring

Flash Professional 8 includes the following new features to help developers create Flash Lite applications:

Flash Lite emulator

The Flash Lite emulator lets you preview your content as it will function on an actual device. The emulator can configure itself to mimic the features available on any supported device. The emulator also provides debugging information that alerts you to potential problems and incompatibilities on the target device.

Device Settings dialog box

You use the Device Settings dialog box to select your test devices and Flash Lite content type. When you test your content in the Flash Lite emulator, you can choose the test device you want the emulator to mimic.

Device document templates

Flash Professional 8 includes document templates to let you quickly start creating content for specific devices and content types.

Workflow for Authoring Flash Lite Applications

The process for creating Flash Lite content is an iterative one that involves the following steps:

Identify your target device(s) and Flash Lite content type

Different devices have different screen sizes, support different audio formats, and have different screen color depths, among other factors. These factors may influence your application's design or implementation.

In addition, different devices support different Flash Lite content types, such as screen savers, stand-alone applications, or animated ring tones. The content type for which you are developing also determines the features that are available to your application.

Create and test your application in Flash

Flash Professional 8 includes a Flash Lite emulator that lets you test your application without having to transfer it to a device. You use the Flash Lite emulator to refine your application design and fix any problems before you test it on a mobile device.

Test the application on your target device or devices

This step is important because the emulator doesn't emulate all aspects of the target device, such as its processor speed, color depth, or network latency. For instance, an animation that runs smoothly on the emulator might not run as quickly on the device, due to its slower processor speed. Or a color gradient that appears smooth in the emulator may appear banded when viewed on the actual device.

After testing your application on a device, you may find that you need to refine the application's design in the Flash authoring tool.

The following figure illustrates the iterative development and testing process described above.



Walkthrough: Café Application Overview

The café application's initial screen contains some introductory text about the restaurant and a menu listing two options: Specials and Reservations. The user selects a menu item by pressing the Up and Down arrows on their device to set the focus, and then pressing the Select key to confirm the selection.



The café application's main screen

If the user selects the Specials menu option, a screen for navigating a list of today's specials appears. To browse images and descriptions of the specials, the user presses the device's Right soft key (labeled Next). To return to the main application, the user presses the Left soft

key (labeled Home).



The cafe application's specials screen

If the user selects the Reservations option on the main screen, the application initiates a phone call to the restaurant. Before Flash Lite dials the requested number, it always asks the user to confirm that they would like to make the call.

Viewing the completed application

A completed version of the café application is the Unit_8 folder. You can view the completed application in the Flash Lite emulator in Flash 8 (If you have the stand-alone version of Flash Lite 1.1 installed on a mobile device, you can transfer the SWF file to your device to view it there.)

To view the completed application in the emulator:

In Macromedia Flash Professional 8, open the file named cafeTutorial_complete.fla located in the /Unit_8/Cafe/ folder .

- Choose **Control > Test Movie** to start the application in the emulator.

To interact with the application, do the following:

- On the main screen, click the Down Arrow key on the emulator's keypad to select the Specials menu item. Then click the Select key on the emulator to view the specials screen.
- On the specials screen, click the Right soft key (Next) on the emulator to view the image and description for each special. Click the Left soft key (Home) to return to the main screen.
- Back on the main screen, select the Reservations menu item to start a phone call to the restaurant.

Creating the Mobile Application

This section contains step-by-step procedures that show you how to recreate the cafe application. The tutorial is divided into three parts:

- Selecting your test devices and content types. In this section, you'll configure your Flash document's publish settings, document settings, and device settings.
- Creating the menu for the application's main screen. From this screen, the user can select from a simple menu to view images and descriptions of the day's specials, or to call the restaurant to make a reservation.
- Creating the specials screen. On this screen, users can press the device's Left soft key to navigate between images and descriptions for each lunch special at the cafe, or press the Right soft key to return to the main screen.

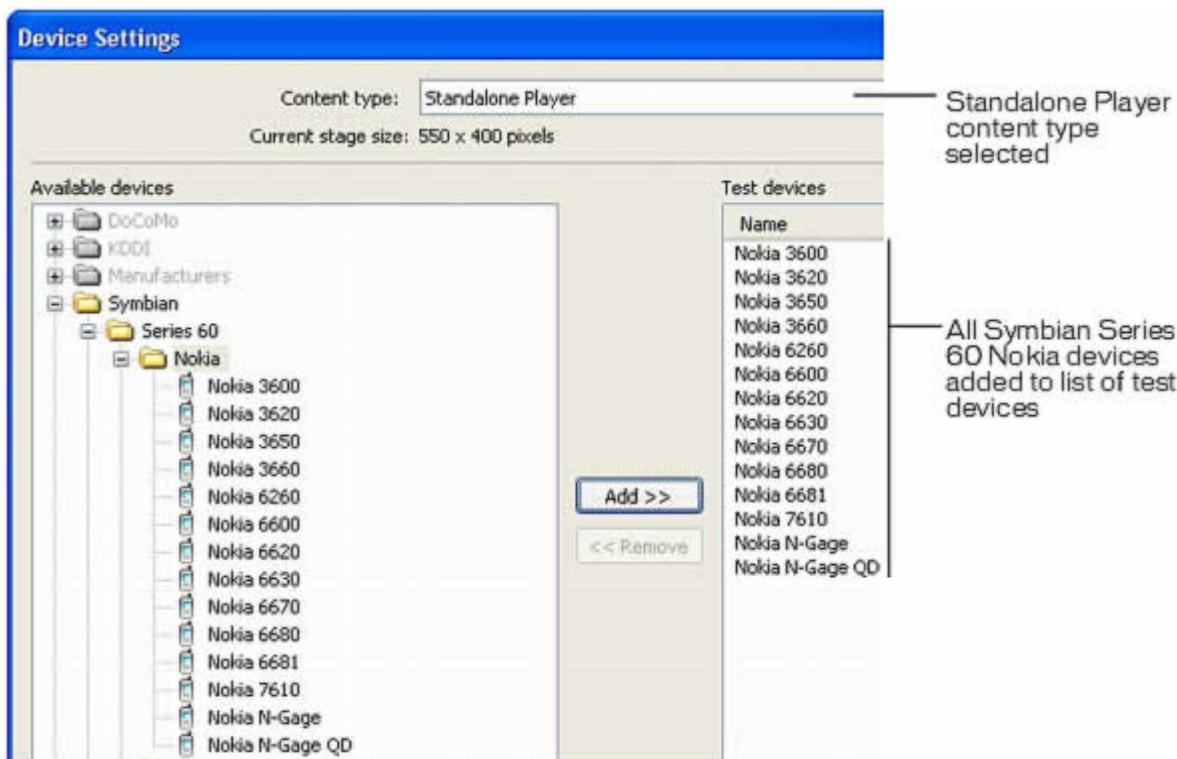
Selecting Your Test Devices and Content Type

In this section, you select the test devices and content type for the application. For this application you'll target the family of Symbian Series 60 devices from Nokia that support the stand-alone version of Flash Lite 1.1.

You use the Device Settings dialog box to select the device(s) and content type that you are targeting. When you test your application in the Flash Lite emulator, the emulator configures itself to match the configuration of the player on the target device as well as the content type.

To select your application's test devices and content type:

1. Open the partially completed source file named **cafe_tutorial_start.fla** located in the **Unit_8** folder.
2. In the Property inspector, click the **Device Settings** button to open the Device Settings dialog box.
3. Select **Standalone Player** from the Content Type pop-up menu.
4. In the list of available devices, double-click the **Symbian** folder to expand it, double-click the **Series 60 folder**, and then double-click the **Nokia** folder.
5. With the **Nokia** folder selected, click **Add** to add all the Nokia Series 60 devices to your list of test devices.
6. When you test your application in the emulator, you'll be able to test your application against any of your test devices for the selected content type.



Click OK to close the Device Settings dialog box.

Save the file as `cafe_tutorial.fla`, or another name of your choosing.

Now that you've selected the test devices and content type, you're ready to start creating the application. In the next section, you'll create the menu for the application's main screen.

Creating the Menu for the Main Screen

In this section, you'll create the menu for the application's main screen. The menu consists of two options: Specials and Reservations. The Specials option takes the user to a screen to view images and descriptions of specials at the cafe. The Reservations option initiates a phone call to the cafe so that the user can make reservations.

The menu consists of two standard Flash buttons that define up, over, and down states. The user gives focus to one of the buttons by pressing the Up or Down Arrow keys on their device. When a button has focus it displays its over state. The button that has focus generates a button press event when the user presses the Select key on the device. This default tab navigation provides an easy way to create a simple user interface for a Flash Lite application.

To create the main screen's menu:

1. In the Timeline window (**Window > Timeline**), select Frame 1 on the menu layer.
2. To create the menu, open the Library panel (**Window > Library**), and drag an instance of the button symbol called **Specials** to the Stage.

Position the button beneath the text field (already in place) that introduces the restaurant.

Drag an instance of the button symbol named **Reservations** to the Stage and position it below the Specials button, as the following image shows:



Select the **Specials button**, and open the Actions panel (**Window > Actions**).

Add the following code to the Actions panel:

```
on (press) {  
    gotoAndStop ("specials") ;  
}
```

This event handler code sends the playhead to the frame labeled **specials** when the user selects this button. You'll create the content for that frame in the next section.

On the Stage, select the **Reservations button** and open the Actions panel again.

In the Actions panel, enter the following code:

```
on (press) {  
    getURL ("tel:1-415-555-1212") ;  
}
```

When the user selects the Reservations menu item, Flash Lite initiates a phone call to the specified number. Flash Lite always prompts the user to allow or deny a request from a SWF file to dial a number.

In the Timeline, select **Frame 1** on the Actions layer.

Open the Actions panel and enter the following code:

```
stop () ;
```

```
focusRect = false;  
_fscommand2 ("resetsoftkeys");  
_fscommand2 ("setquality", "high");  
_fscommand2 ("fullscreen", "true");
```

This code does the following:

- Stops the playhead at this frame.
- Disables the yellow focus rectangle that Flash Lite draws by default around the button or input text field with the current focus.
- Resets the soft keys to their default state. (Later in the tutorial, you'll add code that registers the soft keys for your application to use.)
- Sets the player's rendering quality to high. By default, Flash Lite renders graphical content at medium quality.
- Forces the player to display the application full screen.

To test your work so far, select **Control > Test Movie**.

In the emulator, click the Up or Down Arrow keys on the keypad with your mouse (or press the Up or Down Arrow keys on your computer's keyboard) to give focus to the Specials button.

When the Specials button item gets focus, you will see the button's over state.

Click the **Select** key on the emulator's keypad (or press the Enter key on your keyboard) to select the menu item.

At this point, the specials screen contains no features. In the next section, you'll add interactivity and animation to create the specials screen.

Creating the Specials Screen

In this section, you'll create the user interface elements that let the user browse images and descriptions of each special. The specials screen consists of the following parts:

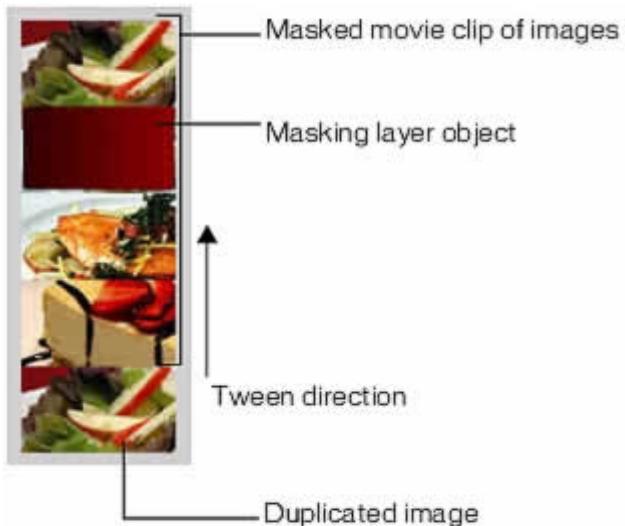
- An animation that transitions between images of each special.
- Dynamic text fields that display the name and description for each special.
- User interface elements that let the user navigate between specials and return to the main application screen.

This section of the tutorial is divided into two parts. In the first part, you'll create the animation that transitions between images of each special. In the second part, you'll add user interface elements and ActionScript to let the user navigate between the images and to display each special's name and description.

Creating the Image Animation

In this section you'll create the tweened animation that transitions between images of each special. When you've completed this section, the animation will play through without stopping. Later in the tutorial, you'll add navigation and ActionScript that lets the user control the animation with the device's Right soft key.

To create the animation you'll use a prebuilt movie clip that contains images of all the specials arranged in a vertical column. You'll use a masking layer to make only one of the images visible. Then you'll create a series of tweens that move the movie clip upward, so that a different image is visible. The last image in the movie clip is a duplicate of the first one, so that the animation sequence can return to its initial state after the user has viewed the final image. The following image illustrates these concepts:



In the last section of the tutorial, you'll add ActionScript and user interface elements that let the user control the animation sequence.

To create the image animation:

1. In the Timeline, select the keyframe on Frame 10 on the layer named Images.
2. Open the Library panel, and drag the symbol named Specials Images movie clip to the Stage.
3. The rest of this tutorial refers to this movie clip simply as the *images movie clip*.
4. With the new movie clip instance selected, set the movie clip's x and y coordinates both to 0 in the Property inspector.
5. This aligns the top-left corner of the images movie clip with the top-left corner of the Stage.

6. On the Images layer, insert keyframes on Frames 20, 30, 40, and 50, as the following image shows:



In the Timeline, select the keyframe on Frame 20.

On the Stage, select the images movie clip, and set its y coordinate to **-100** in the Property inspector.

This moves the movie clip upward on the Stage 100 pixels.

Select the keyframe on Frame 30 in the Timeline, select the images movie clip, and set its y coordinate to **-200** in the Property inspector.

Select the keyframe on Frame 40, select the images movie clip, and set its y coordinate to **-300** in the Property inspector.

Select the keyframe on Frame 50, select the images movie clip, and set its y coordinate to **-400** in the Property inspector.

Select the keyframe on Frame 10, and select Motion from the Tween pop-up menu in the Property inspector.

This tweens the images movie clip's position between the keyframes on Frames 10 and 20.



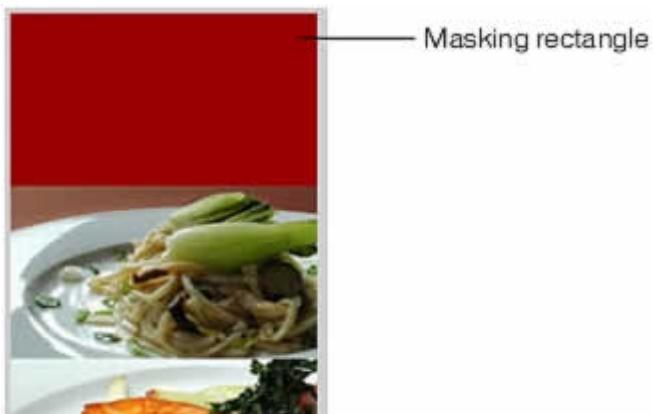
To create transitions between the other images, repeat step 11 for the keyframes located on Frames 20, 30, and 40.

To create the mask layer, select the Images layer in the Timeline, and then select **Insert > Timeline > New Layer** (or click the Insert Layer button in the Timeline).

Insert a keyframe on **Frame 10** of the new mask layer.

Using the Rectangle tool in the Tools palette, create a rectangle over the first (top-most) image in the images movie clip.

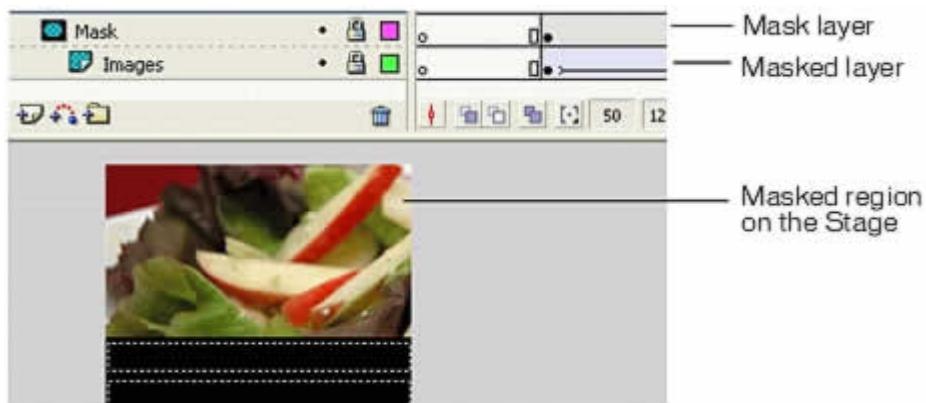
It doesn't matter what fill color you use for the rectangle, but it must be completely opaque.



To make sure the rectangle covers the entire image area, double-click the rectangle to select it, and then use the Property inspector to set its x and y coordinates both to **0**, its width to **176**, and its height to **100**.

Right-click (Windows) or Control-click (Macintosh) the Image Mask layer in the Timeline, and select Mask from the context menu.

The layer is converted to a mask layer, indicated by a mask layer icon. The layer immediately below it is linked to the mask layer, and its contents show through the filled area on the mask.



Save your changes (**File > Save**).

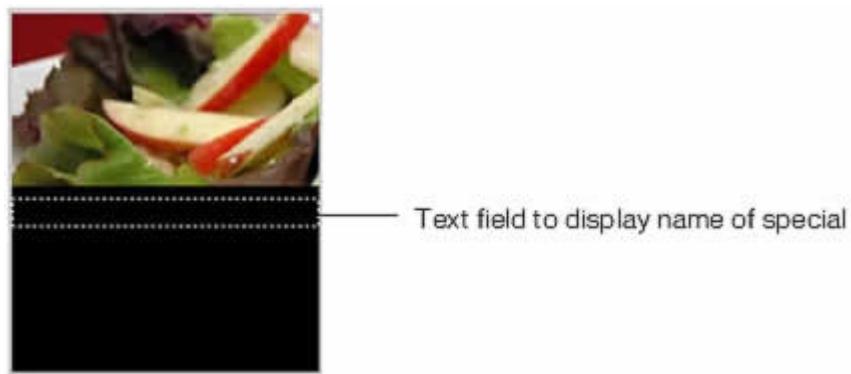
At this point, if you were to test the application in the emulator, the animation you created would play through to the end and then stop.

Adding Navigation and Text to the Specials Screen

In this section, you'll add interactivity to the specials screen that lets the user control the transition between each animation. You'll also add dynamic text fields that display the name and description of each image.

To add text to display the names and descriptions of the specials:

1. In the Timeline, select **Frame 10** on the Text layer.
2. In the Tools palette, select the **Text tool** and create a text field below the first masked specials image.
3. This text field will display the name of the special whose image is currently being displayed.

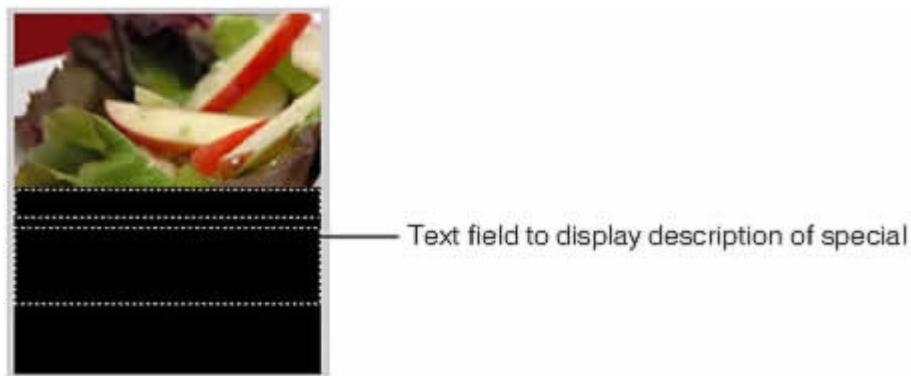


With the text field selected on the Stage, make the following changes in the Property inspector:

- Select **Dynamic Text** from the Text Type pop-up menu.
- Select **Verdana** from the Font pop-up menu.
- Select the **Italics** text style option.
- Set the font size to **10**.
- Select **Bitmap (no anti-alias)** from the Font Rendering Method pop-up menu.
- Type **title** in the Var text box. This is the variable name assigned to the dynamic text field.

Create another text field below the first one to display a short description of the specials being viewed by the user.

Using the Selection tool, resize the text field so that it's about three times as tall as the other text field.



With the text field selected on the Stage, make the following changes in the Property inspector:

- Select **Dynamic Text** from the Text Type pop-up menu.
- Select **Multiline** from the Line Type pop-up menu.
- Select **Verdana** from the Font pop-up menu.
- Set the font size to **10**.
- Select **Bitmap (no anti-alias)** from the Font Rendering Method pop-up menu.
- Type **description** in the Var text box.

In the Timeline, select the keyframe on **Frame 10** on the Actions layer.

Open the Actions panel and add the following code:

```
title = "Summer salad";
description = "Butter lettuce with apples, blood
orange segments, gorgonzola, and raspberry
vinaigrette.";
fscommand2("SetSoftKeys", "Home", "Next");
stop();
```

This code displays the name and description of the special that the user is currently viewing, and stops the playhead. The **SetSoftKeys** command registers the device's soft keys that will let the user return to the home screen, as well as navigate between specials.

On the Actions layer, select the keyframe on **Frame 20** and enter the following code in the Actions panel:

```
title = "Chinese Noodle Salad";
description = "Rice noodles with garlic sauce,
shiitake mushrooms, scallions, and bok choy.";
stop();
```

On the Actions layer, select the keyframe on **Frame 30** and enter the following code in the Actions panel:

```
title = "Seared Salmon";
description = "Filet of wild salmon with
caramelized onions, new potatoes, and caper and
tomato salsa.";
stop();
```

On the Actions layer, select the keyframe on **Frame 40** and enter the following code in the Actions panel:

```
title = "New York Cheesecake";
description = "Creamy traditional cheesecake
served with chocolate sauce and strawberries.";
stop();
```

On the Actions layer, select the keyframe on **Frame 50** and enter the following code in the Actions panel:

```
gotoAndStop("specials");
```

This code returns the playhead to the beginning of the animation sequence. The first and last images in the animation sequence are the same, which creates the illusion of a continuous animation.

Save your changes.

Next you'll add navigation to the specials screen that lets the user navigate between images and descriptions of each special.

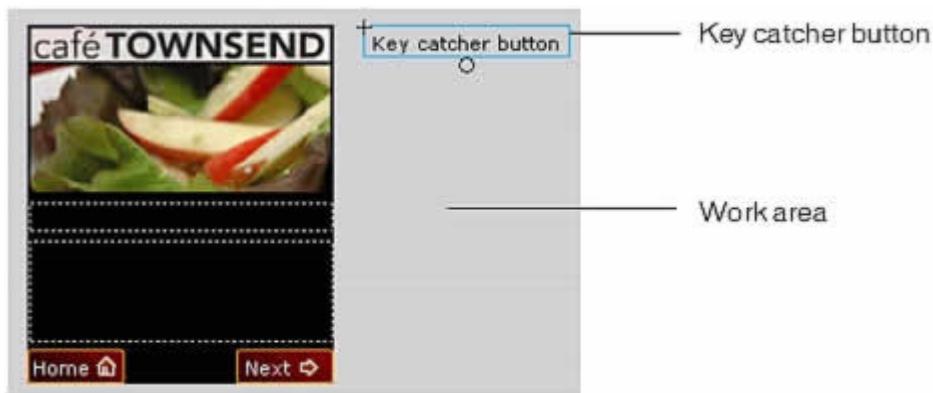
To add navigation to the specials screen:

1. In the Library panel (**Window > Library**), locate the symbol named Home and drag it to the lower-left corner of the Stage.
2. In the Property inspector, set the Home graphic's x coordinate to **0** and its y coordinate to **188**.
3. Drag the symbol named Next from the Library to the lower-right corner of the Stage.
4. In the Property inspector, set the graphic's x coordinate to **120** and its y coordinate to **188**.
5. The Stage in your application should look something like the following screen shot:



In the Timeline, select the keyframe on Frame 10 on the layer named **Key Catcher**.

From the Library, drag the **Key Catcher** button symbol and place it in the work area off the Stage.



1.

The purpose of this button is to "catch" ActionScript keypress events initiated by the user, and then take the appropriate action.

Select the **key catcher** button, and in the Actions panel, enter the following code:

```
// Handle right soft key event ("Next" button):
on(keyPress "<PageDown>") {
    play();
}
// Handle left soft key event ("Home" button):
on(keyPress "<PageUp>") {
    gotoAndStop("main");
}
```

The first **on(keyPress)** handler advances the image animation to the next image in the sequence; the second one sends the playhead to the main application screen.

Choose **Control > Test Movie** to test the final application in the emulator.

LAB: Viewing a Video Game with Flash Lite

Let's try another application to see how the Flash Lite emulator handles the code. In this example, we are going to a simple video game. The Blocks Game is a classic Tetris or Minesweeper type of block game. The idea is to choose blocks that are surrounded and connected to blocks of a similar color. Choosing these blocks removes them from the stage and drops any blocks above them down to fill their empty spaces. The goal is to have as few blocks remaining at the end as possible.

The source file is made up of two frames on the main timeline. One is a welcome screen; the next is the game interface screen. The game interface screen starts by duplicating a template block and randomly changing its properties until the board is full.

You will open up the .FLA file and test the game in the Flash Lite emulator.

To view the completed application in the emulator:

In Macromedia Flash Professional 8, open the file named **blocks.fla** located in the **/Unit_8/Blocks/** folder .

- Choose Control > Test Movie to start the application in the emulator.

To interact with the application, do the following:

- On the main screen, click the Play key on the emulator's keypad to Start the game.
- On the game screen, click the Right, Left, up and Down soft keys on the emulator to change the block selection. Click the Center soft key to start eliminating blocks!

The emulator cannot close the application, so the Done button will not work.